



1 PROCESSING STRUCTURED/HIERARCHICAL CONTENT

2 FIELD OF INVENTION

3 The present invention relates to processing
4 structured/hierarchical content, suitable for processes such
5 as reuse of an annotation and cutout of a Web content.
6 More specifically, the present invention relates to
7 processing the structured/hierarchical content, capable of
8 generating a matching pattern by which the
9 structured/hierarchical content to be subjected to the
10 processing such as the reuse of an annotation and the cutout
11 of a Web content can be detected appropriately.

12 BACKGROUND OF THE INVENTION

13 In recent years, from various viewpoints, attention
14 has been paid to research on highly efficient reuse of
15 portions in Web pages which are present in large amounts and
16 include important contents, by cutting out and converting

1 the portions into individual parts. Note that, in this
2 specification, the term "cutout" is used in meaning for
3 general use by those skilled in the art, and by the
4 "cutout," "cutout" portions are not deleted from a Web
5 content from which the portions are "cut out." Strictly
6 speaking, the "cutout" in this specification is to copy a
7 range of target content portions in an original Web content
8 or the like in order to paste the target content portions to
9 another Web page or the like.

10 In the field of Web services, content cutout has
11 attracted attention as a bridging technology for bridging
12 the existing HTML contents and the Web services. For
13 example, the existing server system can be adapted to the
14 Web services as it is by cutting out, for example, an HTML
15 form for searching an article on a news site and by defining
16 XML input/output to the HTML form.

17 Moreover, in the field of information portals, which
18 aggregate various types of information and provide portal
19 pages coinciding with requests of users, partial components
20 in the existing Web pages are important contents. Regions
21 of top news and headlines are cut out from various news
22 sites and are freely combined, thus making it possible to

1 expand the contents to a great extent. Actually, in the
2 mySiteOutliner, the WebSphere Portal Server or the like, a
3 mechanism for incorporating a part of the existing Web pages
4 into the portal pages is provided as a part of the product.

5 In addition, a standard, which allows a third party to
6 utilize information updated on Web sites and the like by
7 providing the information in an XML form called RSS (Rich
8 Site Summary), has been widespread. At present, the RSS is
9 generated by preparing an exclusive server-side program (CGI
10 and the like). However, if the page cutout technology is
11 used, then conversion of a headline list in a page into the
12 RSS makes it possible to provide a dynamic and highly
13 immediate RSS.

14 Furthermore, in the field of transcoding, a technology
15 has been researched, in which important information in Web
16 pages is submitted preferentially, thus converting the Web
17 pages into pages which are easy for users of pervasive
18 devices and amblyopia users using enlarged browsers to read.
19 A function of conducting page clipping based on annotation
20 description on the XPath base is implemented also in the IBM
21 WebSphere Transcoding Publisher.

22 As described above, it has been known that the part of

1 the Web content can be reused highly efficiently by being
2 cut appropriately. (1) As methods for cutting out the part
3 of the Web pages in the related art, there are two methods,
4 which are: (a) a method using the XPath; and (b) a method
5 using an original tag.

6 (a) Method using XPath:

7 The method using the XPath is a powerful method when
8 the Web pages are assured to be static and unchanged. For
9 example, in the non-patent document 1, the cutout of a
10 content by use of XPath designation is implemented in order
11 to generate pages for portable terminals. However, the
12 designation is troublesome, an application range thereof is
13 narrow, and so on, and therefore, actually, another type of
14 pages for the portable terminals is frequently prepared.
15 Specifically, this method is not actually widespread.
16 Moreover, in the non-patent document 2, a schema is
17 proposed, in which a part of Web pages is selected, and an
18 input portion and an output portion are selected, thus
19 easily enabling the Web pages to be incorporated into the
20 Web services. Although this technology is excellent in that
21 the Web pages can be easily cut out and coupled to the
22 services, the technology involves a problem that it depends

1 on the XPath with regard to the cutout. Furthermore, in the
2 non-patent document 3, a list of images and articles is cut
3 out from the top page of the home page of IBM and the like
4 by use of the XPath, and the cutout list is incorporated
5 into a part of a "personal newspaper." The cutout portions
6 are shifted due to a layout change. Therefore, the shift of
7 the cutout portions is coped with by manually correcting the
8 definition file of the XPath, followed by automatically
9 delivering the cutout portions.

10 (b) Method using original tag:

11 In this method, the original tag is mixed into HTML
12 tags. A particular character string is sometimes designated
13 for an HTML comment. This method is widely used in a portal
14 service such as LYCOS and YAHOO. For example, this method
15 is used for the purpose of displaying an explanation of
16 recommended goods on a shopping page also onto the top page.
17 Because this method can be processed by the simple HTML
18 parser and the like, this method is frequently used in the
19 case of using the HTML parser. This method involves a
20 problem that an original content must be changed.

21 Related arts similar to the present invention will be
22 listed below though they are not the technologies for

1 cutting out the part of the Web page content.

2 (2) Dynamic annotation matching method using XPath set as
3 key (Japanese Patent Application No. 2001-333260 not yet
4 laid-open at the time of preparing this specification):

5 In this method, an XPath included in an annotation is
6 used as a key, and a suitable candidate for the annotation
7 is selected from the plurality of candidates therefor.
8 According to this method, a correct annotation matching has
9 been enabled in many cases by preparing annotations
10 sufficient for covering the entire layout. However, also in
11 many cases, the XPath indicates an incorrect node at an
12 authoring step. As functions for correcting this incorrect
13 node, functions such as an empty content alert, a leaked
14 text alert and a semi-automatic correction of the XPath have
15 been developed. However, in the actual situation,
16 adjustment work is troublesome.

17 (3) Other annotation matching methods:

18 In many cases such as an RDF, the annotations and the
19 pages are matched by use of a collation table and a normal
20 expression of a URL. The present invention greatly differs
21 from these methods in that it performs dynamic matching with
22 the content.

1 (4) Finite difference calculation and use thereof

2 As services/technologies for submitting and reusing
3 only updated information and transmitting a notification
4 mail by use of a finite difference calculation, DiffWeb
5 (example: non-patent document 4), HTML Diff (example:
6 non-patent document 5), MindIt (example: non-patent document
7 6) and the like have been known. In these technologies, a
8 finite difference calculation is performed between a "last
9 past page" and a present page, and a content obtained as the
10 difference is utilized. On the contrary, the present
11 invention is greatly different from these technologies in
12 that an object thereof is to "generate a matching pattern."
13 In addition, in the constituent technologies, the present
14 invention also greatly differ from these technologies in
15 finite difference calculations and statistical processing
16 with past pages in plural versions, a concept of adjacent
17 pages and finite difference calculations therewith, and the
18 like.

19 (5) Simplification technology by finite difference
20 calculation (patent document 1):

21 In this technology, specific information is taken out
22 from the page by use of a finite difference calculation, and

1 the information is simplified. Although this technology is
2 common to the present invention in that adjacent pages are
3 listed up and the finite difference calculations are
4 performed therewith, this technology does not suggest a
5 specific method for cutting out a part of the Web content.

6 (6) Matching technology for a tree structure:

7 As matching technologies for a tree structure by use
8 thereof, a normal expression matching technology (Trex), a
9 matching of the tree structure based on the hedge automaton
10 theory and an application thereof to schema languages (relax
11 and relaxNG) and the like have been researched. These
12 technologies are technologies for searching subtrees (nodes)
13 to be matched with the tree structure on the premise that a
14 matching pattern exists, and do not suggest that they relate
15 to automatic generation of the matching pattern.

16 (7) Technology related to automatic generation of matching
17 pattern:

18 There is a technology called "Examplotron" which
19 automatically generates schema description to be matched
20 with a group of XML samples. This technology is similar to
21 the present invention in that a certain type of matching
22 pattern is automatically generated from a group of XML

1 files. However, this technology is different from the
2 present invention to be described later in that a subject
3 thereof is a group of "well-formatted" XML files "in
4 conformity with a certain tacit schema" and that a strict
5 matching pattern is generated by use of an "embedding
6 structure" of the tags as a key.

7 (8) Efficiency enhancement for work of adding annotations
8 (patent document 2):

9 A common annotation is added to page files analogous
10 to each other in layout structure, and thus an efficiency
11 enhancement for work of adding annotations is attempted. A
12 determination as to whether the page files are analogous in
13 layout structure is performed based on a collation of
14 structural description formulae, and a matching pattern
15 based on statistical information relating to occurrence
16 modes and occurrence frequencies of nodes is not utilized.

17 [Patent document 1]

18 Japanese Patent Laid-Open No. 2002-55872

19 [Patent document 2]

20 Japanese Patent Laid-Open No. 2002-245068

21 [Non-patent document 1]

22 WTP (WebSphere Transcoding Publisher,

1 <http://www-6.ibm.com/jp/software/network/transcoding/>)
2 [Non-patent document 2]
3 CHIP[1] Ito "Construction method of distributed
4 applications by integration of GUI parts and WEB services,"
5 Japan Society for Software Science and Technology WISS 2001
6 Proceedings
7 (<http://ca.meme.hokudai.ac.jp/people/itok/CHIP/indexJ.html>)
8 [Non-patent document 3]
9 IBM mySite Outliner
10 (<http://www-6.ibm.com/jp/pc/clubibm/msol/index.shtml>)
11 [Non-patent document 4]
12 DiffWeb (<http://www.diffweb.com/>)
13 [Non-patent document 5]
14 HTML Diff (<http://www-db.stanford.edu/c3/c3.html>)
15 [Non-patent document 6]
16 MindIt (<http://mindit.netmind.com/mindit.shtml>)

17 SUMMARY OF THE INVENTION

18 It is an aspect of the present invention to provide an
19 apparatus, a method and a program, which exert a great

1 effect when performing processing such as, for example,
2 cutout of a part of structured/hierarchical contents
3 delivered through a network and reuse of an annotation
4 common thereto.

5 It is another aspect of the present invention to
6 provide a processing apparatus for a structured/hierarchical
7 content, a processing method for the structured/hierarchical
8 content and a processing program for the
9 structured/hierarchical content, which are capable of
10 attaining, for example, the cutout of the part of the
11 structured/hierarchical contents and the reuse of the
12 annotation common thereto without using an XPath and adding
13 a tag.

14 In the present invention, in order to identify whether
15 or not contents are the structured/hierarchical contents
16 subjected to the processing such as the partial cutout of
17 the contents and the reuse of the annotation common to a
18 plurality of contents, not an XPath but a matching pattern
19 is used.

20 In the present invention, past and/or adjacent
21 structured/hierarchical contents with respect to a target
22 content are checked, and respective nodes are classified

1 based on statistical information relating to occurrence
2 modes of the nodes in a target subtree and occurrence
3 frequencies of the occurrence modes, thus generating the
4 matching pattern.

5 In an embodiment of a processing apparatus for a
6 structured/hierarchical content of the present invention, it
7 is determined whether or not a structured/hierarchical
8 content delivered through a network includes a content
9 portion matched with a predetermined matching pattern, and
10 if a result of the determination is positive, then
11 predetermined processing is performed for the
12 structured/hierarchical content. Moreover, the processing
13 apparatus for a structured/hierarchical content includes:
14 target subtree setting means for setting a target subtree
15 relating to a range including a target content portion as an
16 extracted portion of the matching pattern in the
17 structured/hierarchical content (hereinafter, referred to as
18 a "target content") from which the matching pattern is to be
19 extracted; occurrence mode detecting means for detecting an
20 occurrence mode of each node of the target subtree by
21 selecting a plurality of past structured/hierarchical
22 contents with respect to the target content and collating

1 the target subtree relating to the target content with a
2 tree relating to each of the past structured/hierarchical
3 contents; statistical information generating means for
4 generating statistical information concerning an occurrence
5 frequency of the occurrence mode of each node in the target
6 subtree based on the plurality of past
7 structured/hierarchical contents; classifying means for
8 performing classification of each node of the target subtree
9 based on the statistical information and a result of
10 detecting the occurrence mode; and matching pattern
11 generating means for generating the matching pattern for the
12 target content portion based on the classification.

13 In a processing method for a structured/hierarchical
14 content of the present invention, it is determined whether
15 or not a structured/hierarchical content delivered through a
16 network includes a content portion matched with a
17 predetermined matching pattern, and if a result of the
18 determination is positive, then predetermined processing is
19 performed for the structured/hierarchical content.

20 Moreover, an embodiment of a processing method for a
21 structured/hierarchical content of the present invention

1 includes: a target subtree setting step of setting a target
2 subtree relating to a range including a target content
3 portion as an extracted portion of the matching pattern in
4 the structured/hierarchical content (hereinafter, referred
5 to as a "target content") from which the matching pattern is
6 to be extracted; an occurrence mode detecting step of
7 detecting an occurrence mode of each node of the target
8 subtree by selecting a plurality of past
9 structured/hierarchical contents with respect to the target
10 content and collating the target subtree relating to the
11 target content with a tree relating to each of the past
12 structured/hierarchical contents; a statistical information
13 generating step of generating statistical information
14 concerning an occurrence frequency of the occurrence mode of
15 each node in the target subtree based on the plurality of
16 past structured/hierarchical contents; a classifying step of
17 performing classification of each node of the target subtree
18 based on the statistical information and a result of
19 detecting the occurrence mode; and a matching pattern
20 generating step of generating the matching pattern for the
21 target content portion based on the classification.

1 BRIEF DESCRIPTION OF THE DRAWINGS

2 For a more complete understanding of the present
3 invention and the advantages thereof, reference is now made
4 to the following description taken in conjunction with the
5 accompanying drawings.

6 Fig. 1 is a constitutional view of a processing system
7 10 for a structured/hierarchical content, with which a Web
8 content processing apparatus 14 is equipped.

9 Fig. 2 is a block diagram of a processing apparatus 18
10 for the structured/hierarchical content.

11 Fig. 3 is a more specific block diagram of matching
12 pattern generating means 30.

13 Fig. 4 is a more specific block diagram of classifying
14 means 29.

15 Fig. 5 is a flowchart of a method for generating a
16 matching pattern based on past structured/hierarchical
17 contents.

18 Fig. 6 is a flowchart of a matching determination
19 method using the matching pattern generated according to the
20 matching pattern generation method of Fig. 5.

21 Fig. 7 is a flowchart portion showing a matching

1 pattern generation step (S51) of Fig. 5 more specifically.

2 Fig. 8 is a more specific block diagram of the
3 classifying means 29.

4 Fig. 9 is a flowchart of a method for generating the
5 matching pattern based on a plurality of
6 structured/hierarchical contents adjacent to a target
7 content.

8 Fig. 10 is a constitutional view of a processing
9 apparatus 74 for the Web content.

10 Fig. 11 is a schematic explanatory view of DP
11 matching.

12 Fig. 12 is a schematic explanatory view in which the
13 DP matching is applied to a difference calculation.

14 Fig. 13 is a view showing a first difference
15 calculation example for a Web content of asahi.com.

16 Fig. 14 is a view showing a second difference
17 calculation example for a Web content of asahi.com.

18 Fig. 15 is an example of a DOM tree.

19 Fig. 16 is a view showing relationships between
20 vectors of serialized nodes and distance vectors at
21 respective stages.

22 Fig. 17 is a view showing the distance vectors at the

1 respective stages in contrast.

2 Fig. 18 is a view showing a Web content having an
3 additional node portion on ends of repeated portions.

4 Fig. 19 is a view showing a Web content having a
5 listing pattern in which bullets are varied.

6 Fig. 20 is a view showing an image of News LYCOS as an
7 example of a Web content including repetitions.

8 Fig. 21 is a view showing an image of a Web content of
9 CNN.COM as an example of the Web content including the
10 repetitions.

11 Fig. 22 is a view showing an image of a Web content in
12 which ten or more tables are continuous in td.

13 Fig. 23 is a view showing an image of an INDEX page of
14 asahi.com and a difference result thereof in contrast.

15 Fig. 24 is a view showing an image of a sports page of
16 asahi.com.

17 Fig. 25 is a view showing a difference result based on
18 the image of Fig. 24.

19 Fig. 26 is a schematic explanatory view of free
20 annotation.

21 Fig. 27 is a schematic explanatory view of fail-safe
22 annotation processing in which already publicly known

1 dynamic matching and the free annotation of Fig. 26 are
2 combined.

3 Fig. 28 is a view showing an anticipated screen of a
4 site pattern analyzer (SPA2) for the free annotation.

5 Fig. 29 is a constitutional view of a matching system
6 in which matching by the matching pattern is incorporated
7 into a dynamic matching method.

8 Fig. 30 is a view showing a result of difference
9 calculation processing for a predetermined region of a
10 certain Web content with adjacent pages.

11 Fig. 31 is a utilization explanatory view of a
12 matching pattern with regard to cutout of numerical values
13 of stock prices from a Web content for stock price
14 information.

15 Fig. 32 is a view showing an example of a Web content
16 where predetermined stationary nodes move.

17 Fig. 33 is a view showing an example of a Web content
18 to be used for partial cutout.

19 Fig. 34 is a view showing a processing course for
20 automatically generating a Web service from the Web content
21 of Fig. 33.

1 DETAILED DESCRIPTION OF THE INVENTION

2 The present invention provides apparatus, methods and
3 programs, which exert a great effect when performing
4 processing such as, for example, cutout of a part of
5 structured/hierarchical contents delivered through a network
6 and reuse of an annotation common thereto. The present
7 invention also provides processing apparatus for a
8 structured/hierarchical content, processing methods for the
9 structured/hierarchical content and processing programs for
10 the structured/hierarchical content. These are capable of
11 attaining, for example, the cutout of the part of the
12 structured/hierarchical contents and the reuse of the
13 annotation common thereto without using an XPath and adding
14 a tag.

15 In order to identify whether or not contents are the
16 structured/hierarchical contents subjected to the processing
17 such as the partial cutout of the contents and the reuse of
18 the annotation common to a plurality of contents, not an
19 XPath but a matching pattern is used.

20 Past and/or adjacent structured/hierarchical contents

1 with respect to a target content are checked, and respective
2 nodes are classified based on statistical information
3 relating to occurrence modes of the nodes in a target
4 subtree and occurrence frequencies of the occurrence modes,
5 thus generating the matching pattern.

6 In an example of a processing apparatus for a
7 structured/hierarchical content of the present invention, it
8 is determined whether or not a structured/hierarchical
9 content delivered through a network includes a content
10 portion matched with a predetermined matching pattern, and
11 if a result of the determination is positive, then
12 predetermined processing is performed for the
13 structured/hierarchical content. Moreover, the processing
14 apparatus for a structured/hierarchical content includes:
15 target subtree setting means for setting a target subtree
16 relating to a range including a target content portion as an
17 extracted portion of the matching pattern in the
18 structured/hierarchical content (hereinafter, referred to as
19 a "target content") from which the matching pattern is to be
20 extracted; occurrence mode detecting means for detecting an
21 occurrence mode of each node of the target subtree by
22 selecting a plurality of past structured/hierarchical

1 contents with respect to the target content and collating
2 the target subtree relating to the target content with a
3 tree relating to each of the past structured/hierarchical
4 contents; statistical information generating means for
5 generating statistical information concerning an occurrence
6 frequency of the occurrence mode of each node in the target
7 subtree based on the plurality of past
8 structured/hierarchical contents; classifying means for
9 performing classification of each node of the target subtree
10 based on the statistical information and a result of
11 detecting the occurrence mode; and matching pattern
12 generating means for generating the matching pattern for the
13 target content portion based on the classification.

14 In an example embodiment of a processing method for a
15 structured/hierarchical content of the present invention, it
16 is determined whether or not a structured/hierarchical
17 content delivered through a network includes a content
18 portion matched with a predetermined matching pattern, and
19 if a result of the determination is positive, then
20 predetermined processing is performed for the
21 structured/hierarchical content. Moreover, the processing
22 method for a structured/hierarchical content of the present

1 invention includes: a target subtree setting step of setting
2 a target subtree relating to a range including a target
3 content portion as an extracted portion of the matching
4 pattern in the structured/hierarchical content (hereinafter,
5 referred to as a "target content") from which the matching
6 pattern is to be extracted; an occurrence mode detecting
7 step of detecting an occurrence mode of each node of the
8 target subtree by selecting a plurality of past
9 structured/hierarchical contents with respect to the target
10 content and collating the target subtree relating to the
11 target content with a tree relating to each of the past
12 structured/hierarchical contents; a statistical information
13 generating step of generating statistical information
14 concerning an occurrence frequency of the occurrence mode of
15 each node in the target subtree based on the plurality of
16 past structured/hierarchical contents; a classifying step of
17 performing classification of each node of the target subtree
18 based on the statistical information and a result of
19 detecting the occurrence mode; and a matching pattern
20 generating step of generating the matching pattern for the
21 target content portion based on the classification.

22 In place of the past structured/hierarchical contents,

1 a plurality of adjacent structured/hierarchical contents can
2 be utilized. The network includes an Intranet, an Extranet
3 and the like as well as the Internet. The
4 structured/hierarchical content is defined as a content
5 including structure information and hierarchy information as
6 well as content itself. As the structured/hierarchical
7 content, for example, there are an XML document and a Web
8 page (HTML file).

9 A processing program for a structured/hierarchical
10 content of the present invention allows a computer to
11 execute the steps of the processing method for a
12 structured/hierarchical content.

13 In order to determine whether or not a
14 structured/hierarchical content to be determined is the
15 structured/hierarchical content adjacent to the target
16 content, analogousness in URL and/or layout is used as a
17 determination factor. In a default state, a system
18 determines overall analogousness while taking the
19 analogousness in the both into consideration. Specifically,
20 the system determines whether or not the
21 structured/hierarchical content to be determined is the
22 structured/hierarchical content adjacent to the target

1 content. For the default as described above, an author can
2 define specific analogousness. Specifically, based on the
3 specific contents of the respective target contents, the
4 author can define specific conditions of the URL and/or
5 layout of the structured/hierarchical content to be
6 determined. Here, the conditions are that the
7 structured/hierarchical content is determined to be the
8 structured/hierarchical content adjacent to the target
9 content. Then, in place of the default, the author can
10 instruct the specific conditions to the computer. The
11 respective means (ex.: the occurrence mode detecting means
12 and the statistical information generating means) and the
13 respective steps (ex.: the occurrence mode detecting step
14 and the statistical information generating step), which
15 determine whether or not the structured/hierarchical content
16 is the structured/hierarchical content adjacent to the
17 target content, implement the determination based on the
18 specific conditions.

19 The "adjacent structured/hierarchical content" can be
20 defined as, though the URL thereof is different from that of
21 the target content, (a) a structured/hierarchical content in
22 which a URL is identical to the URL of the target content in

1 a predetermined ratio or more and/or (b) a
2 structured/hierarchical content in which at least a
3 principal portion of a layout is identical to the layout of
4 the target content. The structured/hierarchical content
5 defined in (b) includes a structured/hierarchical content in
6 which a layout has an identical region to that of the layout
7 of the target content in a predetermined ratio or more.

8 The adjacent structured/hierarchical contents include
9 at least ones listed in the following (a) and (b).

10 (a) Structured/hierarchical content of which directory is
11 common to the target content. A specific example
12 (asahi.com) in the case where the structured/hierarchical
13 content is a Web content is shown as follows.

14 For example, the next URLs are listed as adjacent
15 structured/hierarchical contents to a Web content as the
16 target content of which URL is
17 <http://www.asahi.com/0606/news/national06015.html>.

18 <http://www.asahi.com/0606/news/national06012.html>

19 <http://www.asahi.com/0606/news/national06013.html>

20 <http://www.asahi.com/0606/news/national06014.html>

21 (b) Structured/hierarchical contents in which directories on
22 a predetermined number of hierarchies (for example, two

hierarchies) are common to that of the target content. A specific example (cnn.com) in the case where the structured/hierarchical contents are Web contents is shown as follows.

For example, the next URLs are listed as adjacent structured/hierarchical contents to a Web content as the target content of which URL is

<http://www.cnn.com/2000/US/06/05/sea.based.defense/index.html>.

<http://www.cnn.com/2000/US/06/05/dday.remembrance/index.html>

http://www.cnn.com/2000/US/06/05/helicopter.escape.03/index.
html

<http://www.cnn.com/2000/US/06/05/curbing.terrorism.02/index.html>

qq

A processing apparatus for a structured/hierarchical content automatically generates a matching pattern with high precision only by designating a region desired to be cut out as a method of cutting out a part of a Web page, and

1 realizes robust cutout of an appropriate content. The
2 generation of the matching pattern is performed based on a
3 statistical difference of a Web page with a plurality of
4 pages (hereinafter, a Web content will be referred to as a
5 "page" according to needs). Designated regions (certain
6 nodes on a DOM tree) are compared with a group of past pages
7 stored in advance (subjected to a difference calculation),
8 statistical quantities thereof are calculated, and the nodes
9 are classified into stationary nodes, surely present nodes
10 to be updated and nodes to be added/disappeared. Subtrees
11 subjected to the processing such as detection of an
12 iteration pattern after the classification of the nodes
13 become the matching pattern of the annotation. When the
14 past pages are not present, the matching pattern is obtained
15 in a similar way by performing similar processing performed
16 for adjacent pages. Unlike by the conventional method based
17 on the XPath and the buried tag, the matching pattern as
18 described above does not require changing an original
19 content, and accurate cutout is enabled only by applying the
20 matching pattern as an external annotation. Furthermore,
21 the matching pattern is far more robust in that it is never
22 affected by a change of an upper node even though the change

1 occurs.

2 The "annotation" is defined as predetermined
3 information added to a structured/hierarchical content B
4 when the structured/hierarchical content B is prepared from
5 a predetermined structured/hierarchical content A. This
6 additional predetermined information includes (a)
7 information designating a part of the content A, (b)
8 information concerning portions designated in the content A
9 and/or (c) information obtained by appropriately combining
10 the above (a) and (b). Citing specific examples of the
11 content B, a list summarizing the main items of the content
12 A, which is displayed on the lower side of the content A
13 that is on a screen display mode, and a list of various
14 designation, which includes a change designation of a font
15 size, are added to the content B that is on a screen display
16 mode. The matters thus added are annotations. Upon
17 clicking an item in the list of main items of the added
18 portions, users can jump to a spot in the content B, which
19 corresponds to the portion of the content A. In addition,
20 upon clicking an item in the list of various designations,
21 correspondence processing such as largely displaying fonts
22 on the content B, which includes the portions of the content

1 A, is performed. Note that the matching pattern can be made
2 to function as the annotation by utilizing the matching
3 pattern as information designating the part of the content A
4 and by combining the matching pattern with additional
5 information (information on role and importance of the
6 relevant content portion).

7 Fig. 1 is a constitutional view of the processing
8 system 10 for the structured/hierarchical content, with
9 which the Web content processing apparatus 14 is equipped.
10 A network, to which the present invention is applied, is not
11 limited to the Internet 12 and may be an intranet, an
12 Extranet and the like. The Web content processing apparatus
13 14, the Web clients 15 and the Web servers 16 are connected
14 to the Internet 12 and are constructed to be capable of
15 mutually transmitting and receiving data through the
16 Internet 12. The one Web content processing apparatus 14
17 behaves as a processing apparatus for the
18 structured/hierarchical content, and orders a Web content in
19 accordance with the HTTP (HyperText Transfer Protocol) from
20 corresponding one or a plurality of the Web servers 16 in
21 response to requests from the plurality of Web clients 15.
22 Then, for the Web content, the Web content processing

1 apparatus 14 performs predetermined processing, for example,
2 the impartation of the annotation and/or the cutout of the
3 content, and transmits the processed Web content to the Web
4 clients 15. Note that personal computers serving as the Web
5 clients 15, which are actually operated by the users, may
6 not be directly connected to the Internet 12. The personal
7 computers may be directly connected to an in-company LAN and
8 may be connected to the Internet 12 through a proxy server
9 and a router on the LAN.

10 Fig. 2 is a block diagram of the processing apparatus
11 18 for the structured/hierarchical content. When the
12 structured/hierarchical content to be processed by the
13 processing apparatus 18 are a Web content, the processing
14 apparatus 18 becomes the Web content processing apparatus 14
15 of Fig. 1. An author of the processing apparatus 18 for the
16 structured/hierarchical content prepares an annotation
17 usable commonly to the plurality of structured/hierarchical
18 contents (for example, Web contents), and cuts out
19 predetermined content portions from one or a plurality of
20 the structured/hierarchical contents. The "cutout"
21 mentioned herein does not mean that the content portions are
22 deleted from the structured/hierarchical contents from which

1 the content portions are "cut out," and the relevant cut out
2 portions remain in the structured/hierarchical contents from
3 which the content portions are "cut out." Strictly
4 speaking, the "cutout" mentioned herein is "copying." Then,
5 the author conducts editorial work for the
6 structured/hierarchical contents, such as preparation of new
7 structured/hierarchical contents by pasting the one or
8 plurality of cut out content portions. From a predetermined
9 server of a structured/hierarchical content, the author
10 reads the target content 20 as the structured/hierarchical
11 content, from which the matching pattern is to be extracted,
12 through the network. Then, the author designates a
13 predetermined content portion from the target content 20.
14 The content portion thus designated will be referred to the
15 "target content portion 21." For the target content portion
16 21, the processing apparatus 18 for the
17 structured/hierarchical content automatically sets, as a
18 target subtree, a subtree relating to a range including the
19 target content portion 21 on the DOM tree of the target
20 content 20. The target subtree is required relating to the
21 range including the target content portion 21. It is
22 preferable to set the range at a range as small as possible,

1 and the range may be set at a range of the content portion,
2 which is somewhat larger than the target content portion 21.
3 Prior to the editorial work at this time, the author
4 notifies the XPath of the target content 20 to the
5 structured/hierarchical content database 26 in advance (for
6 example, one week before, 10 days before and one month
7 before this editorial work). After the notification, the
8 structured/hierarchical content database 26 automatically
9 and periodically accesses contents relating to the target
10 content 20, and stores the contents therein. Hence, in the
11 case of this user's work for the target content 20, the
12 sufficient number of past structured/hierarchical contents
13 involved in the target content 20 are stored in the
14 structured/hierarchical content database 26. The occurrence
15 mode detecting means 27 reads out the past
16 structured/hierarchical contents involved in the target
17 content 20 from the structured/hierarchical content database
18 26 one by one or in a lump, collates the target subtree
19 relating to the target content portion 21 with trees
20 relating to the past structured/hierarchical contents, and
21 detects the occurrence mode of the respective nodes of the
22 target subtree. Preferably, the plurality of past

1 structured hierarchical contents involved in the target
2 content 20 are structured/hierarchical contents stored
3 within a predetermined past period from the present point of
4 time, that is, from the time of generation processing of the
5 matching pattern. Note that the target content 20 and the
6 past structured/hierarchical contents involved in the target
7 content 20 have the same URL (Uniform Resource Locator).
8 The statistical information generating means 28 generates
9 statistical information concerning the occurrence
10 frequencies of the occurrence modes of the respective nodes
11 in the target subtree based on the plurality of past
12 structured/hierarchical contents. The classifying means 29
13 classifies the respective nodes of the target subtree based
14 on the result of detecting the occurrence modes in the
15 occurrence mode detecting means 27 and the statistical
16 information generated by the statistical information
17 generating means 28.

18 The processing in the occurrence mode detecting means
19 27, the statistical information generating means 28 and the
20 classifying means 29 will be described more specifically.
21 In the occurrence mode detecting means 27, the target
22 subtree relating to the target content 20 is collated with

1 one tree of the past structured/hierarchical contents.
2 Thus, the respective nodes of the target subtree can be
3 classified into (N1) nodes that occur also in the
4 structured/hierarchical contents and have the same contents
5 as those of the structured/hierarchical contents, (N2) nodes
6 that occur also in the structured/hierarchical contents but
7 have different contents, and (N3) nodes that do not occur in
8 the structured/hierarchical contents. Note that each
9 content of the nodes means a description content between a
10 start tag and an end tag in the XML as the
11 structured/hierarchical content. The occurrence mode
12 detecting means 27 collates the trees of the predetermined
13 plural number of past structured/hierarchical contents with
14 the target subtree, thus making it possible to detect the
15 statistical information concerning the occurrence
16 frequencies of (N1) and (N2). The statistical information
17 generating means 28 generates this statistical information.
18 The classifying means 29 has preset threshold values V1 and
19 V2 for the frequencies at which the nodes occur in the modes
20 of (N1) and (N2). Typically, V1 and V2 are equal to each
21 other ($V1=V2$), however, V1 and V2 may be values different
22 from each other. Typically, both of V1 and V2 are set equal

1 to 70% ($V1=V2=70\%$). A specific example of the node
2 classification in the classifying means 29 is as follows.
3 The nodes in the mode of (N1), of which occurrence
4 frequencies are equal to/larger than $V1$ ($\geq V1$), are
5 classified into stationary nodes. The nodes in the mode of
6 (N2), of which occurrence frequencies are equal to/larger
7 than $V2$ ($\geq V2$), are classified into updated nodes. Nodes
8 that are not classified into either the stationary nodes or
9 the updated nodes are classified into additional nodes.

10 The matching pattern generating means 30 generates the
11 matching pattern based on the classification results in the
12 classifying means 29. Matching processing between the
13 matching pattern generated in the matching pattern
14 generating means 30 and the content portion will be
15 described later in detail with reference to Fig. 6.

16 Fig. 3 is a more specific block diagram of the
17 matching pattern generating means 30. The repeated portion
18 detecting means 34 detects repeated portions in the target
19 subtree based on the classification into the stationary
20 nodes, the updated nodes and the additional nodes. The
21 repeated information-added matching pattern generating means
22 35 generates a matching pattern including presence

1 information of the repeated portions. In such a way, even
2 if the structured/hierarchical content determined whether it
3 is matched with the generated matching pattern repeats the
4 repeated portions arbitrary times, the matching pattern
5 allows the structured/hierarchical content to be usable as
6 one matched with the matching pattern.

7 Fig. 4 is a more specific block diagram of the
8 classifying means 29. In order to improve a layout during
9 display, the structured/hierarchical content sometimes
10 includes an image for a spacer and a bullet image. The
11 image for the spacer corresponds to a "spacer GIF" of an
12 HTML file. The plurality of images are used for one
13 structured/hierarchical content in order to secure a blank
14 region, and designated sizes thereof are mutually different.
15 Meanwhile, the bullet image corresponds to a "bullet image"
16 of the HTML file. The plurality of bullet images are used
17 for one structured/hierarchical content. Sizes thereof are
18 designated to be identical, or no designation is made
19 thereto. The formed-for-spacer image detecting means 38
20 detects whether or not the nodes of the target subtree are
21 nodes relating to the images for the spacer. The bullet
22 image detecting means 39 detects whether or not the nodes of

1 the target subtree are nodes relating to the bullet images.
2 The first classifying means 40 classifies the nodes relating
3 to the images for the spacer into the additional nodes. The
4 second classifying means 41 allocates the nodes relating to
5 the bullet images to the same classification among those of
6 the stationary nodes, updated nodes and additional nodes
7 even if display contents thereof are mutually different.
8 The classification outputting means 42 includes a function
9 of summarizing the classifications of the nodes by the first
10 and second classifying means 40 and 41, and generates the
11 output of the classifying means 29.

12 The processing apparatus 18 for the
13 structured/hierarchical contents of Fig. 2 generates the
14 matching pattern based on the past structured/hierarchical
15 contents (contents of which URLs are the same as that of the
16 target content) with respect to the target content.
17 However, the processing apparatus 18 can also generate the
18 matching pattern based on the structured/hierarchical
19 contents adjacent to the target content. The generation of
20 the matching pattern based on the adjacent
21 structured/hierarchical contents may be implemented (a) only
22 when there are no past content portions with respect to the

1 target content or (b) regardless of the existence of the
2 past content portions with respect to the target content.
3 For example, the business article page on the home page of
4 Asahi Shimbun (www.asahi.com) includes a date in the URL as
5 follows, and can be browsed together with up-to-date
6 business articles for a predetermined period including the
7 present. Note that, in the example below, the business
8 article is dated as of October 19.

9 "http://www.asahi.com/business/update/1019/002.html"

10 In order to generate an appropriate matching pattern even in
11 the case as described above, the present invention
12 introduces a concept titled "adjacent
13 structured/hierarchical contents to a target content." The
14 adjacent structured/hierarchical contents are
15 structured/hierarchical contents, which have URLs analogous
16 to that of the target content and are made to belong to the
17 same group as that of the target content in the case of a
18 matching determination by means of the matching pattern.
19 The analogous range of the URLs is varied depending on the
20 extent to which the author determines that differences of
21 the structured/hierarchical contents are allowable and the
22 different contents belong to the same group. The URLs

1 include directories (portions partitioned by forward slashes
2 (/) in the example of the business article in Asahi Shimbun)
3 in the respective hierarchies. When the URLs of the
4 contents to be determined whether or not they are the
5 adjacent structured/hierarchical contents are collated with
6 the URL of the target content, if directories up to a
7 predetermined number (one or more) of hierarchies from the
8 uppermost hierarchy are identical and only directories in
9 lower hierarchies from the hierarchies where the directories
10 are identical, the content portions subjected to the
11 determination may be determined as adjacent content
12 portions. Specific examples of the adjacent content
13 portions are listed as follows. In the next cases, the
14 structured/hierarchical contents subjected to the
15 determination are determined as the adjacent
16 structured/hierarchical contents.

17 (a) Only a portion recognized as a date in the URL differs
18 from that of the target content. In the above-described
19 example of the business article of Asahi Shimbun, the
20 relevant portion is "1019."

21 (b) Only a portion used as numbering in the URL differs from
22 that of the target content. In the above-described example

1 of the business article of Asahi Shimbun, the relevant
2 portion is "002.html."

3 (c) Only the above-described (a) and (b) differ from those
4 of the target content.

5 In the case where the processing apparatus 18 for the
6 structured/hierarchical contents of Fig. 2 generates the
7 matching pattern based on the adjacent
8 structured/hierarchical contents in place of the past
9 structured/hierarchical contents, only a different point
10 from the case where the processing apparatus 18 generates
11 the matching pattern based on the past
12 structured/hierarchical contents will be described. The
13 structured/hierarchical content database 26 stores the
14 plurality of structured/hierarchical contents adjacent to a
15 predetermined structured/hierarchical content in advance in
16 order to cope with a selection of an arbitrary
17 structured/hierarchical content as the target content 20 of
18 this time by the author. The occurrence mode detecting
19 means 27 reads out the structured/hierarchical contents
20 adjacent to the target content 20 one by one or in a lump
21 from the structured/hierarchical content database 26,
22 collates the target subtree relating to the target content

1 29 with the trees relating to the respective
2 structured/hierarchical contents adjacent to the target
3 subtree relating to the target content 20, and detects the
4 occurrence modes of the respective nodes of the target
5 subtree. The statistical information generating means 28
6 generates the statistical information concerning the
7 occurrence frequencies of the occurrence modes of the
8 respective nodes of the target subtree based on the
9 plurality of adjacent structured/hierarchical contents. The
10 classifying means 29 classifies the nodes of the target
11 subtree based on the result of detecting the occurrence
12 modes in the occurrence mode detecting means 27 and the
13 statistical information generated by the statistical
14 information generating means 28. In the case of using the
15 adjacent structured/hierarchical contents in place of the
16 past structured/hierarchical contents, the processing in the
17 occurrence mode detecting means 27, the statistical
18 information generating means 28 and the classifying means 29
19 will be described as follows. In the occurrence mode
20 detecting means 27, the target subtree relating to the
21 target content 20 is collated with the tree of one adjacent
22 structured/hierarchical content. Thus, the respective nodes

1 of the target subtree can be classified into (N1) nodes that
2 occur also in the structured/hierarchical contents and have
3 the same contents as those of the structured/hierarchical
4 contents, (N2) nodes that occur also in the
5 structured/hierarchical contents but have different
6 contents, and (N3) nodes that do not occur in the
7 structured/hierarchical contents. The occurrence mode
8 detecting means 27 collates the tree of each of the
9 predetermined plural adjacent structured/hierarchical
10 contents with the target subtree, thus making it possible to
11 detect the statistical information concerning the occurrence
12 modes of (N1) and (N2) for each node of the target subtree.
13 The statistical information generating means 28 generates
14 this statistical information. The classifying means 29 has
15 preset threshold values V1 and V2 for the frequencies at
16 which the nodes occur in the modes of (N1) and (N2).
17 Typically, V1 and V2 are equal to each other ($V1=V2$),
18 however, V1 and V2 may be values different from each other.
19 Typically, both of V1 and V2 are set equal to 70%
20 ($V1=V2=70\%$). A specific example of the node classification
21 in the classifying means 29 is as follows. The nodes in the
22 mode of (N1), of which occurrence frequencies are equal

1 to/larger than $V1$ ($\geq V1$), are classified into the stationary
2 nodes. The nodes in the mode of (N2), of which occurrence
3 frequencies are equal to/larger than $V2$ ($\geq V2$), are
4 classified into the updated nodes. Nodes that are not
5 classified into either the stationary nodes or the updated
6 nodes are classified into the additional nodes.

7 Note that the matching pattern generating means 30 of
8 Fig. 3 and the classifying means 29 of Fig. 4 are also
9 applied in the case of generating the matching pattern based
10 on the adjacent structured/hierarchical contents in place of
11 the past structured/hierarchical contents.

12 Fig. 5 is a flowchart of a method for generating the
13 matching pattern based on the past structured/hierarchical
14 contents. The agent of the respective steps of the matching
15 pattern generation method is a computer installed with a
16 program for executing the respective steps of the matching
17 pattern generation method. This computer corresponds to the
18 Web content processing apparatus 14 in the example of Fig.
19 1. In S46, the target subtree is set. From a predetermined
20 structured/hierarchical content server, the author reads the
21 target content 20 as the structured/hierarchical contents,
22 from which the matching pattern is to be extracted, through

1 the network. Next, the author designates a predetermined
2 content portion from the target content 20. In S46, for the
3 target content portion 21, a subtree including the range of
4 the target content portion 21 is automatically set as a
5 target subtree on the DOM tree of the target content 20.
6 The target subtree is required relating to the range
7 including the target content portion 21. It is preferable
8 to set the range at a range as small as possible, and the
9 range may be set at a range of the content portion, which is
10 somewhat larger than the target content portion 21. In S47,
11 the past structured/hierarchical contents with respect to
12 the target content portion 20 are read out one by one or in
13 a lump from the structured/hierarchical content database 26.
14 In S48, the target subtree relating to the target content 20
15 is collated with the trees relating to the past
16 structured/hierarchical contents, and thus the occurrence
17 modes of the respective nodes of the target subtree are
18 detected. Preferably, the plurality of past
19 structured/hierarchical contents with respect to the target
20 content 20 are structured/hierarchical contents within a
21 predetermined past period from the present point of time,
22 that is, from the time of generation processing of the

1 matching pattern. Note that the target content 20 and the
2 past structured/hierarchical contents with respect to the
3 target content 20 have the same URL (Uniform Resource
4 Locator). In S49, the statistical information concerning
5 the occurrence frequencies of the occurrence modes of the
6 respective nodes in the target subtree is generated based on
7 the plurality of past structured/hierarchical contents. In
8 S50, the respective nodes of the target subtree are
9 classified based on the result of detecting the occurrence
10 modes in the occurrence mode detecting means 27 and the
11 statistical information generated by the statistical
12 information generating means 28.

13 The processing in S48, S49 and S50 will be described
14 more specifically. In S48, the target subtree relating to
15 the target content 20 is collated with one tree of the past
16 structured/hierarchical contents. Thus, the respective
17 nodes of the target subtree can be classified into (N1) the
18 nodes that occur also in the structured/hierarchical
19 contents and have the same contents as those of the
20 structured/hierarchical contents, (N2) the nodes that occur
21 also in the structured/hierarchical contents but have
22 different contents, and (N3) the nodes that do not occur in

1 the structured/hierarchical contents. In S48, the tree of
2 each of the predetermined plural number of past
3 structured/hierarchical contents is collated with the target
4 subtree, thus making it possible to detect the statistical
5 information concerning the occurrence frequencies of (N1)
6 and (N2) for each node of the target subtree. In S50, the
7 preset threshold values V1 and V2 for the frequencies at
8 which the nodes occur in the modes of (N1) and (N2) are
9 provided. Typically, V1 and V2 are equal to each other
10 ($V1=V2$), however, V1 and V2 may be values different from
11 each other. Typically, both of V1 and V2 are set equal to
12 70% ($V1=V2=70\%$). A specific example of the node
13 classification in S50 is as follows. The nodes in the mode
14 of (N1), of which occurrence frequencies are equal to/larger
15 than V1 ($\geq V1$), are classified into the stationary nodes.
16 The nodes in the mode of (N2), of which occurrence
17 frequencies are equal to/larger than V2 ($\geq V2$), are
18 classified into the updated nodes. The nodes that are not
19 classified into either the stationary nodes or the updated
20 nodes are classified into additional nodes.

21 In S51, the matching pattern is generated based on the
22 classification result in S50. Fig. 6 is a flowchart of a

1 matching determination method using the matching pattern
2 generated according to the matching pattern generation
3 method of Fig. 5. In S55, a content portion (hereinafter,
4 referred to as a "determined content portion") to be
5 determined from now on whether it is matched with the
6 matching pattern is read out. In S56, it is determined
7 whether or not the determined content portion has a portion
8 matched with the matching pattern. The determined content
9 portion when being determined to be matched with the
10 matching pattern may be located at an arbitrary position in
11 a structured/hierarchical content (hereinafter, referred to
12 as a "determined content") including the relevant determined
13 content portion. Specifically, the determined content
14 portion matched with the matching pattern is correctly
15 determined to be matched with the matching pattern even if
16 the determined content portion is located at the arbitrary
17 position of the determined content. If a result of the
18 determination in S56 is positive, then the processing
19 proceeds to S57, and otherwise, this method is ended. In
20 S57, predetermined processing is implemented for the
21 determined content portion. For example, the predetermined
22 processing is (a) association of related information with a

1 content portion of a determined content and (b) copy
2 processing for a determined content portion of a determined
3 content in order to utilize the content portion of the
4 determined content for another structured/hierarchical
5 content (those skilled in the art call the processing
6 "cutout"). The related information of (a) is, for example,
7 an annotation.

8 Fig. 7 is a flowchart portion showing the matching
9 pattern generation step (S51) of Fig. 5 more specifically.
10 In S60, the repeated portions in the target subtree are
11 detected based on the classification into the stationary
12 nodes, the updated nodes and the additional nodes. In S61,
13 a matching pattern including presence information of the
14 repeated portions detected in S60 is generated. In such a
15 way, even if the structured/hierarchical content determined
16 whether it is matched with the generated matching pattern
17 has portions repeated arbitrary times, the generated
18 matching pattern allows the structured/hierarchical content
19 to be usable as one matched with the matching pattern.

20 Fig. 8 is a more specific block diagram of the
21 classifying means 29. In Fig. 8, the series of S64 and S65
22 and the series of S66 and S67 are illustrated so as to be

1 processed in parallel. However, these series may be
2 serially processed such that one of the series precedes the
3 other. In S64, it is detected whether or not the nodes of
4 the target subtree are nodes relating to the images for the
5 spacer. In S65, the nodes relating to the images for the
6 spacer are classified into the additional nodes. In S66, it
7 is detected whether or not the nodes of the target subtree
8 are nodes relating to the bullet images. In S67, the nodes
9 relating to the bullet images are allocated to the same
10 classification among those of the stationary nodes, updated
11 nodes and additional nodes even if the bullet images display
12 different contents. In S68, the classification results in
13 S65 and S67 are summarized and outputted.

14 Fig. 9 is a flowchart of a method for generating the
15 matching pattern based on the plurality of
16 structured/hierarchical contents adjacent to the target
17 content. With reference to Fig. 5, the method for
18 generating the matching pattern based on the past
19 structured/hierarchical contents with respect to the target
20 content has been described. Meanwhile, the generation
21 method described with reference to Fig. 9 may be implemented
22 (a) only when there are no past content portions with

1 respect to the target content or (b) regardless of the
2 existence of the past content portions with respect to the
3 target content. A different point of the flowchart of Fig.
4 9 from the flowchart of Fig. 5 is that S47b to S50b are
5 implemented in place of S47 to S50. Only the different
6 point will be described below.

7 In S47b, the structured/hierarchical contents adjacent
8 to the target content 20 are read out one by one or in a
9 lump from the structured/hierarchical content database 26.
10 In S48b, the target subtree relating to the target content
11 20 is collated with the trees relating to the
12 structured/hierarchical contents adjacent the target
13 subtree, and the occurrence modes of the respective nodes of
14 the target subtree are detected. In S49b, the statistical
15 information concerning the occurrence frequencies of the
16 occurrence modes of the respective nodes in the target
17 subtree is generated based on the plurality of adjacent
18 structured/hierarchical contents. In S50b, the respective
19 nodes of the target subtree are classified based on the
20 result of detecting the occurrence modes in the occurrence
21 mode detecting means 27 and the statistical information
22 generated by the statistical information generating means

1 28. The processing in S48b, S49b and S50b will be described
2 more specifically. In S48b, the target subtree relating to
3 the target content 20 is collated with the tree of one
4 adjacent structured/hierarchical content. Thus, the
5 respective nodes of the target subtree can be classified
6 into the (N1) nodes that occur also in the
7 structured/hierarchical contents and have the same contents
8 as those of the structured/hierarchical contents, the (N2)
9 nodes that occur also in the structured/hierarchical
10 contents but have different contents, and the (N3) nodes
11 that do not occur in the structured/hierarchical contents.
12 In S48b, the tree of each of the predetermined plural
13 adjacent structured/hierarchical contents is collated with
14 the target subtree, thus making it possible to detect the
15 statistical information concerning the occurrence modes of
16 (N1) and (N2) for each node of the target subtree. In S49b,
17 this statistical information is generated. In S50b, the
18 threshold values V1 and V2 preset for the frequencies at
19 which the nodes occur in the modes of (N1) and (N2) are
20 acquired. Typically, V1 and V2 are equal to each other
21 ($V1=V2$), however, V1 and V2 may be values different from
22 each other. Typically, both of V1 and V2 are set equal to

1 70% ($V1=V2=70\%$). A specific example of the node
2 classification in S50b is as follows. The nodes in the mode
3 of (N1), of which occurrence frequencies are equal to/larger
4 than $V1$ ($\geq V1$), are classified into the stationary nodes.
5 The nodes in the mode of (N2), of which occurrence
6 frequencies are equal to/larger than $V2$ ($\geq V2$), are
7 classified into the updated nodes. Nodes that are not
8 classified into either the stationary nodes or the updated
9 nodes are classified into the additional nodes. In S51, the
10 matching pattern is generated based on the classification
11 result in S50.

12 Note that the flowcharts of Figs. 7 and 8 are also
13 applied in the case of generating the matching pattern based
14 on the adjacent structured/hierarchical contents in place of
15 the past structured/hierarchical contents.

16 [Example]

17 In Example, a Web content is selected as the
18 structured/hierarchical content. A matching pattern of a
19 content statistically calculated by use of a result of a
20 difference calculation between a past page and an adjacent
21 page is used for specifying a cutout portion. Fig. 10 is a
22 constitutional view of the processing apparatus 74 for the

1 Web content. The Web client 76, the transcoding module 77
2 and the Web server 78 are connected to the Internet and
3 constructed to be capable of mutually transmitting and
4 receiving data. The user 75 operates the Web client 76 and
5 requests the transcoding module 77 to send the transcoded
6 HTML 81 thereto. Upon receiving the request from the Web
7 client 76, the transcoding module 77 receives the target
8 HTML 79 from the corresponding Web server 78, transcodes the
9 target HTML 79 based on an annotation from the annotation
10 database, and sends the transcoded HTML 81 to the Web client
11 76. Note that, though the annotation database is typically
12 equipped in a computer packaged with the transcoding module
13 77, the annotation database may be located at a separate
14 place from the transcoding module 77 and may be connected to
15 the transcoding module 77 through the Internet. The
16 annotation editor 85, the cache database 86 and the site
17 pattern analyzer 88 are packaged or equipped in the computer
18 equipped with the annotation database. The cache database
19 86 is equipped with a mechanism of caching calculation
20 algorithms of adjacent pages and past pages in plural
21 versions and with a function of acquiring a page of a
22 designated URL by periodically touring the page. The cache

1 database 86 prepares an annotation of each target HTML 79 by
2 use of the annotation editor 85. In order to improve the
3 work efficiency of the annotation author 84, reuse of the
4 annotation, in which the same annotation is commonly used
5 for the plurality of target HTMLs 79, is performed. In
6 order to achieve appropriate reuse of the annotation, a
7 plurality of similar target HTMLs 79 are collected into one
8 group, and the same annotation set is used for each group.
9 Note that the annotation set is one formed by collecting the
10 plurality of annotations. Whether or not the target HTMLs
11 79 belong to a predetermined group is determined by
12 collating the target HTMLs 79 with a predetermined matching
13 pattern.

14 The matching pattern can be used for the purpose of
15 realizing an "annotation matched with a content though the
16 annotation may occur in any portion in the page." Thus,
17 robust cutout against a change of a layout can be realized.
18 In the following, a method for automatically generating the
19 matching pattern by differences with the adjacent pages and
20 the past pages, which is a basic method, will be first
21 described. Then, an example on an actual user interface
22 will be described.

1 [Occurrence frequency calculation in past page based on
2 difference calculation]

3 As a premise, the difference calculation used herein
4 is one equivalent to that used in the simplification by the
5 difference calculation. Even by using an algorithm
6 performing a strict difference calculation of an XML, such
7 as XMLDiff, the method in this example is executable. Here,
8 as shown in Fig. 11, a method is used, which calculates a
9 longest common node string (LCNS) by use of DP matching
10 after once DOM trees are serialized. Although this method
11 cannot perform an accurate difference calculation for the
12 tree, this method is suitable for the method in this example
13 because it has already confirmed that no practical problem
14 is involved therein, the calculation is fast, it is easy to
15 control an element to be calculated, and so on. Description
16 will be made below on the assumption that this method is
17 used for the difference calculation. In addition, in many
18 processing steps that follow, "common nodes" are used as a
19 result of the difference calculation. The "common nodes"
20 are a group of nodes common to two DOM trees and can be
21 obtained by selecting portions other than the differences
22 from the difference calculation result. In the difference

1 calculation method by the DP matching, which is used this
2 time, common portions can be obtained as the LCNS halfway
3 during the calculation. Therefore, the common nodes can be
4 obtained without actually calculating difference portions.
5 Accordingly, though the difference calculation is not
6 actually conducted halfway of the entire calculation, the
7 method in this example can be generally grasped as a
8 variation of the difference calculation. Therefore, in the
9 following description, the notation "difference calculation"
10 is used. Strictly, a "group of common nodes (LCNS) as a
11 result of the difference calculation" is used.

12 Fig. 11 is a schematic explanatory view of DP
13 matching. For example, the first and second inputs are
14 defined as "KWPSIKAWNA" and "ABPSAWNDS," respectively. By
15 the DP matching, "PSAWN" as a longest common node string
16 (LCNS) of these inputs is outputted. In the DP matching,
17 even if excessive elements ("IK" of the first input in the
18 example) are interposed in the DOM tree, if relative orders
19 of the elements are identical, then a string formed of these
20 elements can be extracted as the LCNS.

21 Fig. 12 is a schematic explanatory view in which the
22 DP matching is applied to the difference calculation. The

1 target portions of the DOM trees of the target page and the
2 compared page (past page or adjacent page) are inputted to
3 the serializing means 91 and 92, respectively, and
4 arrangements thereof are converted from a tree type to a
5 serial type. The DP matching means 93 calculates the
6 longest common node string (LCNS) based on inputs from the
7 serializing means 91 and 92. The LCNS removing means 94 as
8 differentiating means outputs the difference DOM tree as a
9 value obtained by subtracting the LCNS from the DOM tree of
10 the target page.

11 • Type A: Calculation of matching pattern in case where past
12 page is present

13 A state is considered, where an annotation author has
14 already designated a certain node group on a DOM tree by use
15 of an annotation editor.

16 Step 1: A target subtree is decided. One ancestor node
17 commonly owned by a group of subject nodes is searched. A
18 <body> node is commonly owned in any case, and therefore, it
19 is obvious that the node as described above is essentially
20 present.

21 Step 2: A list of the past pages is acquired from a cache.
22 It is desirable that the annotation author stores the past

1 pages during the period spanning from several days to
2 several weeks in advance. As the numbers of the past pages
3 are more, it is possible to generate a more robust pattern.
4 Step 3: Difference calculations between the past pages and a
5 page becoming a target at present are performed (difference
6 calculations for the first time). In the case of performing
7 serialization for the difference calculations, the entire
8 elements in the designated group are added to the subject of
9 serialization. Node rows selected by the DP matching are
10 only "stationary nodes." "In the case where important
11 attributes regarding appearance and function coincide" in
12 checking identifications of tags, the tags are determined to
13 be the same. This is because there is a possibility that a
14 page author adds attributes different in detailed points to
15 tags having the same appearance and functions. In the
16 implementation of this example, the identifications were
17 determined by attributes to be described below. Depending
18 on subjects, for example, in the case where an src tag of
19 img is completely controlled by a load balancing system of
20 Akamai Technologies, Inc. or the like, the src tag will not
21 be subjected to the identification determination.
22 Base: "class," "id," "name," "style," "width," "height,"

1 "bgcolor"
2 img series: "alt," "src"
3 link series: "href"
4 form series: "action," "method," "type," "value"
5 table series: "align," "valign," "rowspan," "colspan,"
6 "size," "color," "face"

7 In the above, the "attribute regarding the appearance" is
8 one regarding an appearance of an HTML file in a displayed
9 state, such as "bgcolor." The "attribute regarding the
10 function" is one that does not affect the displayed state of
11 the HTML file, such as "href" of the link series, and
12 "action" and "method" of the form series.

13 Step 4: A frequency at which each node in the tree of the
14 target group occurs in the past pages is calculated as a
15 "stationary index." For example, now, in the case where the
16 target group is compared with twelve past pages and a
17 certain element occurs in eight pages thereof, the
18 stationary index becomes 0.67 ($=8/12$). Not only such a
19 simple percentage but also any number can become an index as
20 long as it is a numerical value indicating the frequency.

21 Step 5: Nodes determined not to be the stationary nodes are
22 classified into "essential/updated nodes" (essentially

1 occurring and being updated) and "additional nodes" (that
2 may be added/deleted to be varied) by a difference
3 calculation for the second time. The "essential/updated
4 nodes" will be abbreviated as "updated nodes" in this
5 specification according to needs. In Step 3, only in the
6 case where the character string is completely matched with
7 the text node, both were defined as identical. In this
8 step, in "the case where a text node (image element) is
9 present" even if a character string or an image is not
10 matched therewith, both are determined to be identical.
11 Moreover, anchor(a) elements of the both are determined to
12 be identical even if the href attributes do not coincide
13 with each other. ones having the src attribute such as
14 iframe and the href attribute are processed in a similar
15 way. Nodes that are not included in the node list of Step 2
16 but included in a node list in this step can be said to be
17 "nodes that essentially occur and are always updated (text,
18 anchor, image)."
19 Step 6: The frequency of each node listed up in Step 5 is
20 calculated. This index is similar to that in Step 3, and a
21 simple percentage can also be used therefor.
22 Step 7: The respective nodes are classified into the

1 stationary nodes, the updated nodes and the additional nodes
2 based on the results of Steps 4 and 6. Such classification
3 is performed by determining the index by means of a certain
4 threshold value. For example, when the stationary index
5 exceeds 70%, the node is determined to be a stationary node.
6 However, among the target subtrees subjected to the
7 calculation in Step 1, in all of node groups that are not
8 designated by the annotation author (node groups that are
9 not included in subtrees extended to leaf directions with
10 the subject node group in Step 1 as a root node), "any" is
11 set in "pat: type attribute."

12 The results of the difference calculations as
13 described above are shown in Figs. 13 and 14. Figs. 13 and
14 14 show examples of difference calculations for Web contents
15 of asahi.com. Figs. 13(a) and 14(a) shown originals
16 (original content portions, and Figs. 13(b) and 14(b) show
17 results of the difference calculations. Colored background
18 portions in Figs. 13(b) and 14(b) are portions of stationary
19 nodes, and white background portions are portions of updated
20 text nodes. It is understood that the character string of
21 "zenbun (full text)>>" in Fig. 13 and the character string
22 of "saishinnyu-su (up-to-date news) can be determined to be

1 regular.

2 Step 8: Furthermore, types of images are determined in order
3 to improve precision. This is performed for the purpose of
4 determining bullets in the list and "spacer GIFs" for
5 securing blank regions and rejecting bullets and spacer GIFs
6 from the iteration patterns. The plurality of spacer GIFs
7 are used for one page, and are images different in
8 designated size for each time when being used. The
9 plurality of bullet images are used for one page, and are
10 images always used in the same size or without designation
11 of its size. Next, the iteration of the subtree in the
12 pattern is analyzed. Some methods are present for analyzing
13 the iteration pattern of the subtrees, and here, an
14 algorithm is shown, where the detection of the iteration
15 pattern is performed at a relatively high speed by searching
16 a vector obtained by serializing the subtrees.

17 Step 9: The classified tree structures are serialized, the
18 following information is calculated for each node, and thus
19 a new vector is generated.

20 Distance vector: distance on a vector where subtrees which
21 occur next and have "the same level, the same tag type and
22 the same value node" are serialized.

1 For example, an example as below is considered. Here, the
2 updated node is written as: pat:type="updated," and the
3 additional node is written as: "pat:type="inserted"
4 <div>
5
6 <pat:text pat:type="updated"/>
7 <pat:text pat:type="updated"/>
8 <pat:text pat:type="updated"/>
9 <imag src="../../new.gif" pat:type="inserted">
10
11
12
13 <pat:text pat:type="updated"/>
14 <pat:text pat:type="updated"/>
15 <pat:text pat:type="updated"/>
16
17
18 <pat:text pat:type="updated"/>
19 <pat:text pat:type="updated"/>
20 <pat:text pat:type="updated"/>
21
22 </div>

1 Fig. 15 is an example of the DOM tree. In this
2 example, nodes corresponding to the elements div, ul and li
3 are stationary nodes, and nodes in the lowermost layer are
4 updated text nodes or additional image nodes. Fig. 16 shows
5 relationships between vectors of serialized nodes and
6 distance vectors at respective stages. Fig. 16(a) shows a
7 serialized vector of the nodes, and Figs. 16 (b) to (f) show
8 distance vectors at the first, second, third and fourth
9 stages, respectively. Note that this serialization is
10 serialization of depth-preferential system. In the
11 conversion from the DOM tree of Fig. 15 to the vector of
12 Fig. 16(a), the "additional node (pat:type="inserted") is
13 not incorporated in the serialized vector. Thus, a
14 temporarily inserted content can be rejected from the
15 calculation of the pattern, and the robustness of the
16 pattern can be enhanced. For example, also in the pattern
17 illustrated in Fig. 18, the portion shown in the drawing can
18 be rejected as an "additional node portion" from the
19 iteration determination. The additional node is included in
20 the pattern by subsequent processing.

21 In addition, even if the images determined to be the
22 bullet images in Step 8 are mutually different, these images

1 are determined to be identical. Thus, for example, a
2 listing pattern as shown in Fig. 19, in which the bullets
3 are varied, can also be detected as an iteration pattern.

4 Furthermore, a "distance vector at the second stage"
5 indicating a "distance to the second identical node" is
6 calculated (Fig. 16(d)). In a similar way, distance vectors
7 at the third stage (Fig. 16(e)) and the fourth stage (Fig.
8 16(f)) are sequentially calculated, and the number of stages
9 is increased until the value (number) of all the nodes
10 becomes one-third ($1/3$) or more of the vector length. This
11 is because one iteration of the longest iteration pattern
12 becomes one-third ($1/3$) or less of the vector length. In
13 the example of the drawing, the vector length is 22 nodes,
14 and therefore, it is not necessary to calculate distance
15 vectors at stages after the fourth stage (Fig. 16(f)).

16 Step 10: An iteration pattern is detected based on the
17 vectors calculated in Step 9. Specifically, "a portion
18 where the same distance is repeated twice or more" in the
19 distance vector is searched. For example, in the case where
20 the distance "5" is repeated, when the total distance
21 exceeds 10, the iteration pattern is detected. The reason
22 of the above operation is that the same element pattern is

1 repeated three times or more.

2 In the example of Fig. 17, patterns are detected
3 across the first stage and the third stage. In this case,
4 the patterns may be included in the distance vectors at the
5 second and third stages. However, in this case, checking is
6 made such that the iteration pattern "does not bridge
7 across" the subtrees. For example, in the case where a DOM
8 structure to be described below is present, checking is made
9 such that ranges from 6 to 10 and from 11 to 15 are not
10 detected but ranges from 8 to 12 and from 13 to 17 are
11 detected as iterations. Specifically, a distance of an
12 iteration of lower nodes is adapted not to be detected
13 across an iteration of upper nodes.

14 1:

15 2:

16 3: keizai (economy)

17 4:

18 5:

19 6: <pat:text pat:type="updated"/>

20 7:

21 8:

22 9: <pat:text pat:type="updated"/>

```

1  10:   <li><pat:text pat:type="updated"/></li>
2  11:   <li><pat:text pat:type="updated"/></li>
3  12:</ul>
4  13:<ul>
5  14:   <li><pat:text pat:type="updated"/></li>
6  15:   <li><pat:text pat:type="updated"/></li>
7  16:   <li><pat:text pat:type="updated"/></li>
8  17:</ul>
9  Step 11: The detected repeated portions are enclosed by
10 <repeat> tags, and the iteration is removed. With regard to
11 the repeated portions, in addition to the portions where the
12 identical distances ("7" in Fig. 17) continue, a portion
13 corresponding to the last of the iteration is added to the
14 pattern. Furthermore, the inserted nodes rejected during
15 the serialization in Step 7 are inserted into the
16 corresponding positions.
17 <div>
18     <repeat>
19         <ul>
20             <li><pat:text pat:type="updated"/></li>
21             <li><pat:text pat:type="updated"/></li>
22             <li><pat:text pat:type="updated"/>

```

```
1             
2             </ls>
3         </ul>
4     </repeat>
5 </div>
6 Step 12: The classified tree structure is shaped as a
7 pattern for matching. An output example of this algorithm
8 will be shown. For the sake of convenience, not the
9 existing pattern matching description but an original
10 expression in which a few tags are added to the html
11 description will be used in the following description. This
12 is because readability of the algorithm is considered, and
13 the algorithm can be converted into the existing language
14 equivalent thereto in description capability. This will be
15 described later. Figs. 20 and 21 show images of Web
16 contents of News LYCOS and CNN.COM as examples of Web
17 contents including the iterations, respectively. Moreover,
18 Fig. 22 shows an image of a Web content in which ten or more
19 tables are continuous in td. Patterns (in XML format)
20 automatically generated from these Web contents will be
21 shown below. The base tag set accords with xhtml, and tags
22 for the pattern are inserted therein as pat name spaces.
```

1 Note that, in the Web content of Fig. 21, the pattern (in
2 XML format) automatically generated on the assumption that
3 two tables are selected from among the ten or more
4 continuous tables is shown.

5 In addition, a notation is used here, which expresses
6 the iteration and the like by means of a prefix "pat" by
7 utilizing the name space, and however, this notation is set
8 equivalently replaceable with another normal tree
9 expression/description. For example, TREX for use in
10 relaxNG has description power sufficient for the pattern in
11 this method, and is usable for the pattern description of
12 this method. This will be described later.

13 Pattern (in XML format) automatically generated from the Web
14 content of Fig. 20

```
15 <table width="168">
16   <tbody>
17     <tr bgcolor="dedede">
18       <td>
19         <b>
20           <span>topics</span>
21         </b>
22       </td>
```

```
1      </tr>
2      <pat:repeat>
3          <tr bgcolor="ffffff">
4              <td>
5                  <small>
6                      <a>
7                          <pat:text pat:type="updated">
8                              </a>
9                          </small>
10                     </td>
11                 </tr>
12             </pat:repeat>
13         <tr bgcolor="ffffff">
14             <td>
15                 <small>
16                     <div align="right">
17                         <span>[</span>
18                         <a>
19                             <span>motto-miru (see more)</span>
20                         </a>
21                         <span>]</span>
22                     </div>
```

```

1         </small>
2     </td>
3 </tr>
4 </tbody>
5 </table>
6 Pattern (in XML format) automatically generated from the Web
7 content of Fig. 21
8 <table width="345">
9     <tbody>
10         <tr>
11             <td bgcolor="#CC0000" style="background-color: #c00;">
12                 <span class="cnnMainHeaderBarText" style="color:
13 #fff">
14                     <b>
15                         <span>?AMERICA AT HOME?</span>
16                     </b>
17                 </span>
18             </td>
19             <td bgcolor="#000033" style="background-color: #003;"
20 width="60%" align="right">
21                 <span class="cnnMainHeaderBarText">
22                     <a style="color: #fff">

```

```
1          <b>
2          <span>more>></span>
3          </b>
4          </a>
5          <span>?</span>
6          </span>
7      </td>
8  </tr>
9  <tr>
10     <td colspan="2">
11         <div class="cnnMainT2List">
12     <!--investigation -->
13         <pat:repeat>
14             <div style="padding-top: 3px; padding-bottom:
15 3px; ">
16                 <li>
17                     <span class="cnnMainT2Area">
18                         <a>
19                             <pat:text pat:type="any">
20                                 </a>
21                             </span>
22                         </li>
```



```
1         </div>
2     </pat:repeat>
3     <div style="padding-top: 3px; padding-bottom: 3px;">
4         <li>
5             <span class="cnnMainT2Area">
6                 <a>
7                     <pat:text pat:type="any">
8                         </a>
9                 </span>
10            </li>
11        </div>
12    <div style="padding-top: 3px; padding-bottom: 3px;">
13        <li>
14            <span class="cnnMainT2Area">
15                <span>Fact Sheet: </span>
16                <a>
17                    <pat:text pat:type="any">
18                        </a>
19                </span>
20            </li>
21        </div>
22 <!-- /investigation -->
```

```

1      </div>
2      </td>
3      </tr>
4      </tbody>
5  </table>
6  Pattern (in XML format) automatically generated from the Web
7  content of Fig. 22
8  <td width="99%">
9      <pat:element pat:type="any">
10     <table width="100%" pat:type="targetnode">
11         <tbody>
12             <tr bgcolor="dedede">
13                 <td>
14                     <b>
15                         <span>keizai (economy)</span>
16                     </b>
17                     <small>
18                         <pat:text pat:type="any">
19                     </small>
20                 </td>
21                 <td align="right">
22                     <small>

```

```

1      <a>
2          <span>keizai (economy)</span>
3      </a>
4      <span> | </span>
5      <a>
6          <span>kigyo (enterprise)</span>
7      </a>
8      <span> | </span>
9      <a>
10         <span>market</span>
11     </a>
12 </small>
13 </td>
14 </tr>
15 </tbody>
16 </table>
17 <table width="100%" pat:type="targetnode">
18     <tbody>
19         <tr>
20             <td>
21                 <a>
22                     <b>

```

```

1          <pat:text pat:type="any">
2          </b>
3          </a>
4          <small>
5          <nobr>
6          <pat:tex pat:type="any">
7          </nobr>
8          </small>
9      </td>
10 </tr>
11 <tr>
12 <td>
13          <pat:text pat:type="any">
14          <nobr>
15          <pat:text pat:type="any">
16          <a>
17          <pat:text pat:type="any">
18          </a>
19          <pat:text pat:type="any">
20          </nobr>
21          <nobr>
22          <pat:text pat:type="any">

```

```
1      <a>
2          <pat:text pat:type="any">
3      </a>
4          <pat:text pat:type="any">
5      </nobr>
6  </td>
7 </tr>
8 </tbody>
9 </table>
10 <pat:element pat:type="any">
11 </td>
12 Type B: Calculation of matching pattern when past page is
13 not present
14 The case where the past page is not present occurs not only
15 when the caching of the past pages is not performed but also
16 when the URLs are generated every day. For example, in the
17 case where a date is utilized as a part of a URL as in a URL
18 of a newspaper article, it is obvious that no past pages can
19 be present
20 (http://www.asahi.com/international/update/1005/010.html).
21 Moreover, no past pages can be present either in the case of
22 a search result page query or the like. In such a case, a
```

1 concept of "adjacent pages" is introduced. The adjacent
2 pages are a group of pages having conditions as below.

3 (a) The URLs are mutually analogous. The analogousness of
4 the URLs is defined by an edit distance between the URLs.

5 Example:

6 Target: [http://www.asahi.com/international/update/1005/010.](http://www.asahi.com/international/update/1005/010.html)
7 html

8 Adjacent URL:

9 <http://www.asahi.com/international/update/1005/012.html>

10 (b) Layouts are mutually analogous. For this determination,
11 a clustering technology by comparison of table structures of
12 the layouts is utilized (Example: the above-mentioned Patent
13 Document 2). This technology is a method for clustering the
14 layouts of the respective pages by use of the embedding
15 structures of the tables as a base, and by the technology, a
16 list of the pages mutually analogous in layout can be
17 obtained.

18 A group of pages that applies to these conditions is
19 the "adjacent pages." Processing steps therefor will be
20 described below. In a similar way to Type A, a state is
21 considered, where the annotation author has already
22 designated a certain node on the tree by use of the

1 annotation editor.

2 Step 1: The list of the adjacent pages is acquired. It
3 is assumed that a cache server has a calculation
4 algorithm of the adjacent pages, and the list of the
5 adjacent pages is acquired from the cache server. Not
6 only the present adjacent pages but also the past
7 adjacent pages are acquired.

8 Step 2: Difference calculations between the respective
9 adjacent pages and a page becoming a target at present
10 are performed. In a similar way to Step 3 of Type A,
11 in the case of performing serialization for the
12 difference calculations, the identifications of the
13 text nodes and the image elements are defined by which
14 "the character strings and the images are completely
15 identical."

16 Step 3: Frequencies at which the respective nodes in
17 the tree of the target group occur in the past pages
18 are calculated as "stationary indices."

19 Step 4: In the case where the "text nodes (image
20 elements) are present" even if the character strings
21 and images are not matched between the adjacent pages
22 and the target page, both are determined to be

1 identical, and the difference calculations therebetween
2 are performed. Nodes that are not included in the node
3 list of Step 2 but included in a node list in this Step
4 can be said to be "texts (images) essentially occurring
5 and being always updated."

6 Step 5: The frequencies of the nodes listed up in Step
7 4 are calculated. Indices of these are similar to
8 those of Step 3, and simple percentages can also be
9 used therefor.

10 Step 6: The respective nodes are classified into the
11 stationary nodes, the updated nodes and the additional
12 nodes based on the results of Steps 3 and 5.

13 Such classification is performed by determining the
14 indices by means of a certain threshold value. For example,
15 when stationary indices exceed 70%, the nodes are determined
16 to be stationary nodes. Examples of the results are shown
17 in Figs. 23 to 25. Figs. 23(a) and 23(b) show an image of
18 the INDEX page of asahi.com and a difference result thereof
19 in contrast. Fig. 24 shows an image of the sports page of
20 asahi.com, and Fig. 25 shows the difference result based on
21 the image of Fig. 24. Actual results of the difference

1 calculations are displayed on a color screen on which areas
2 occurring on a larger number of adjacent pages are displayed
3 deeper blue. In Fig. 24, fixed items in the index list are
4 made stationary. In Fig. 23(b), areas of the items of
5 "weather," "society"..., and "this morning paper" and of the
6 buttons on the left of the items are detected as the
7 stationary nodes of which color is deeper blue though they
8 are difficult to see because the color images in actual are
9 displayed monochrome. Moreover, in Fig. 25, the background
10 of the area including the body text of the article is
11 displayed whitish gray, and it is seen that the body text of
12 the article is detected as one to be updated.

13 Processing from here is similar to that subsequent to
14 Step 8 of Type A. The greatest difference between Type A
15 and Type B is the number of pages to be compared. In Type
16 A, there are certain comparison objects that are the past
17 pages. Therefore, the nodes can be classified appropriately
18 by comparing a few pages. On the contrary, in Type B, the
19 difference calculations must be performed for the adjacent
20 pages, that is to say, objects that are "uncertain" and
21 "involve a possibility that layouts thereof are essentially
22 different." Therefore, it is desirable that the indices be

1 calculated as statistical quantities obtained by performing
2 the difference calculations with an order from several
3 hundred pages to several thousand pages if possible.

4 Next, various utilization modes of the matching
5 pattern generated by the present invention will be
6 described.

7 • Free annotation:

8 Free annotation is a method for matching a concerned
9 group with a certain pattern even if the group occurs
10 anywhere in a page without the XPath (or only by roughly
11 detecting a position of the group). Fig. 26 is a schematic
12 explanatory view of the free annotation. In Fig. 26, the
13 same elements as those in Fig. 10 are denoted by the same
14 reference numerals, and description thereof will be omitted.
15 The user 75 issues a transmission request of the
16 predetermined accessible HTML 96 to the transcoding module
17 77. The transcoding module 77 receives the corresponding
18 target HTML 79 from the corresponding Web server 78, and
19 requests the entire annotations to be associated with the
20 target HTML 79 to the annotation database. Each in the
21 annotation database and the annotation set 97 has a matching
22 pattern corresponding to an annotation indicating a specific

1 group. The annotation database selects the annotation set
2 97 having a matching pattern matched with each subtree of
3 the target HTML 79, and sends the matching pattern to the
4 transcoding module 77. The transcoding module 77 sends, to
5 the Web client, the accessible HTML 96 prepared by
6 transcoding the target HTML 79 based on the annotation set
7 97 received from the annotation database. In the
8 transcoding module 77, robust designation of a cutout
9 position of the target HTML 79 can be realized when the
10 target HTML 79 is transcoded. Moreover, in the case of
11 using the free annotation for the transcoding, the free
12 annotation can be used for the application purpose such as a
13 detection of a group moving in the page or a detection of a
14 group matched with a certain pattern from among the entire
15 pages in a certain site. This free annotation is performed
16 after the conventional dynamic matching method, leading to a
17 possibility that an annotation can be added to the leaked
18 text or a page with which the annotation is not matched.
19 Thus, a fail-safe system can be constructed.

20 Fig. 27 is a schematic explanatory view of fail-safe
21 annotation processing in which the already publicly known
22 dynamic matching and the free annotation of Fig. 26 are

1 combined. In Fig. 27, portions corresponding to those in
2 Figs. 10 and 26 are added with the same reference numerals,
3 and description thereof will be omitted. At the first
4 stage, in the dynamic matching, the transcoding module 77
5 searches an annotation set in which the entire annotations
6 are matched with the target HTML 79 with regard to the
7 XPath. If the annotation set is present, then annotations
8 thereof are set to the transcoding module 77. The
9 transcoding module 77 transcodes the target HTML 79 based on
10 the annotation set, prepares the transcoded HTML 81, and
11 sends the transcoded HTML 81 to the Web client 76. If, in
12 the dynamic matching, the annotation set to be matched
13 cannot be searched in the annotation database 99 for the
14 dynamic matching, then the transcoding module 77 issues an
15 instruction of the free annotation to the annotation
16 database, and receives the annotation set 97 from the
17 annotation database 80. Subsequently, the transcoding
18 module 77 transcodes the target HTML 79 based on the
19 annotation set 97, prepares the transcoded HTML 81, and
20 sends the transcoded HTML 81 to the Web client 76.

21 In this method, stationarity of the tree is calculated
22 by use of the statistical method. Therefore, there are

1 limitations that it is difficult to calculate a "series
2 (group) of nodes of which positions are greatly changed on a
3 DOM tree for each page." For example, it is thought that
4 such a group of nodes does not occur frequently enough to be
5 expressed as a statistical quantity in the case where a
6 certain table can occur on any place every time when it is
7 reloaded. Accordingly, the "free group" detectable by use
8 of this method is premised on that "there is a default
9 position that is not greatly varied," and has limitations in
10 this point. However, as cases where such an annotation
11 shift occurs, it is experientially known that frequencies of
12 "new tr is inserted to cause the shit," "sequences of tr are
13 replaced" and the like are high. This method is
14 sufficiently effective in that it can cope with the changes
15 described above.

16 [Free annotation utilization example: Preparation of free
17 annotation by annotation editor]

18 The following is an operation order of the annotation editor
19 by the author.

20 Step 1: Selecting an arbitrary region (subtree of the

1 DOM tree) by the annotation editor.
2 Step 2: Instructing an addition of a new group.
3 Step 3: Checking a check box of "free annotation" in a
4 group definition dialog, followed by automatic
5 calculation of a matching pattern by the system.
6 Step 4: determining applicability of the matching
7 pattern of Step 3 to another page by the user (author)
8 using the annotation editor.

9 [Free annotation utilization example: Correction of
10 annotation by site pattern analyzer for free annotation]

11 The free annotation requires a management application
12 similar to the conventional site pattern analyzer. Fig. 28
13 shows an anticipated screen view of the site pattern
14 analyzer (SPA2) for the free annotation. The URLs are
15 arrayed on the left side of the annotation matching window,
16 the free annotations are arrayed on the horizontal
17 coordinate, and the matchings with the respective pages are
18 displayed. It is possible to sort the pages by clicking the
19 numbers of annotations. When the author discovers a pattern
20 mistakenly matched, the author corrects the pattern by steps

1 as below.

2 Step 1: Selecting a plurality of URLs correctly matched.

3 Step 2: Selecting the plurality of URLs mistakenly matched.

4 Subsequently, the system corrects the matching pattern so as
5 to be matched with the entire URLs correctly matched and not
6 to matched with the group of URLs mistakenly matched.

7 Application to conventional dynamic matching:

8 This method can be applied as content conditions to be added
9 with the XPath to the conventional dynamic matching method.

10 Fig. 29 is a constitutional view of a matching system in
11 which the matching by the matching pattern is incorporated
12 into the dynamic matching method. In Fig. 29, the same
13 components as those in Fig. 26 are denoted by the same
14 reference numerals, and description thereof will be omitted.
15 In the annotation database 101, with regard to the target
16 HTML 79, the matching by the matching pattern is also
17 determined in addition to the matching by the XPath.
18 Consequently, the determination precision is enhanced. Note
19 that, in the respective annotation sets of the annotation
20 database 101, the painted means annotations matched with
21 both of the XPath and the matching pattern.

22 [Application example to conventional dynamic matching:

1 Addition of group matching as detailed conditions to group
2 by annotation editor]

3 The operation procedure of the author is as follows.

4 Step 1: Selecting an arbitrary region (a subtree of the
5 DOM tree) by the annotation editor. This operation is
6 not different from the standard one.

7 Step 2: Instructing an addition of a new group.

8 Step 3: Pushing a "detailing" button in an auto-group
9 definition dialog.

10 Following this operation, the system automatically
11 calculates the matching pattern. In a standard PC (personal
12 computer), it takes several second to several ten seconds to
13 calculate the matching pattern for Type A, and it takes
14 several ten second to several minutes to calculate the
15 matching pattern for type B.

16 Step 4: determining applicability of the matching
17 pattern to another page by the author using the
18 annotation editor.

19 [Application example to conventional dynamic matching:
20 Application to dynamic matching annotation by site pattern

1 analyzer]

2 The operation procedure of the author is as follows.

3 Step 1: Searching a group mistakenly matched by the
4 site pattern analyzer.

5 Step 2: Selecting several pages correctly matched and
6 several pages mistakenly matched. This operation is
7 similar to that of semi-automatic detailing.

8 Step 3: Selecting a group of pages formed by the pages
9 correctly matched from the list, and selecting the
10 "detailing" therefor.

11 Step 4: Automatically generating a matching pattern
12 with which the group of correct pages is essentially
13 matched by use of the difference calculation.

14 Step 5: Confirming that the generated matching pattern
15 is not matched with the error group. In the case where
16 the matching with the error group occurs, the
17 conditions are further detailed by use of the
18 conventional semi-automatic correction function of the
19 XPath.

20 Next, the precision in the case of using the adjacent
21 pages will be described. In the case of using the adjacent

1 pages for the purpose of generating the matching pattern,
2 there is a problem that the generated matching pattern is
3 greatly varied depending on the listed-up adjacent pages.
4 Fig. 30 shows a result of difference calculation processing
5 for a predetermined region of a certain Web content with the
6 adjacent pages. Fig. 30(a) shows a target Web content for
7 which the matching pattern is to be obtained. Fig. 30(b)
8 shows a result of detecting types of nodes by the difference
9 calculation. In Fig. 30(b), the background of the region of
10 "kanren-joho (related information)" has a thin color
11 similarly to the background of headline regions changed
12 according to needs. The character string of "kanren-joho is
13 obviously stationary and should be incorporated into the
14 matching pattern. However, in the case of performing the
15 differences with the adjacent pages, it is difficult to
16 determine such variations of the place and a large character
17 string. The present invention copes with this problem by
18 two methods.

19 (a) Strict selection of the adjacent pages. Only pages
20 considered to use the same layout are listed up by use
21 of the above-mentioned clustering technology for the
22 layout.

1 (b) Interface for error correction. The
2 above-mentioned site pattern analyzers SPA and SPA2
3 have interfaces for correcting such errors.

4 Determination of types of cutout information
5 Fig. 31 is a utilization explanatory view of a matching
6 pattern with regard to cutout of numerical values of stock
7 prices from a Web content for stock price information. Fig.
8 31(a) shows a Web content submitting stock price
9 information, and Fig. 31(b) shows stationary nodes detected
10 by the difference calculation with the past pages. It is
11 also thought that the cutout of the numerical values of the
12 stock price from the table of the stock price information
13 and the like is incorporated into the matching pattern of
14 the annotations. For example, from text of "12-ji 13-pun
15 koshin (updated at 12: 13), it is possible to cut out time
16 information of HH and MM by description of:
17 <pat:data pat:typ="date" pat:format="updated at hour HH
18 minute MM" pat:xpath="table[1]/tr[1]/td[3]/text()[1]"/>.
19 As described above, it is also possible to incorporate the
20 cutout portions of numerical value data and text data into
21 the matching pattern. It is thought that, in such a way,

1 there is a great effect for conversing the data into the
2 RSS, WSXL or VoiceXML.

3 Fusion with method of dynamic annotation/Utilization of
4 fast algorithm matched with XPath set:

5 It is also possible to grasp the matching of the
6 subtree of this time as matching of an XPath set. In such a
7 way, it is possible to utilize the method of the fast
8 matching of the XPath set, which has been proposed
9 heretofore. However, the iterations using repeat and
10 pat:type="inserted" cannot be expressed, and therefore, the
11 entire matching patterns cannot be converted.

12 (Using the XPath of the group as a root)

```
13    /tr[1]
14    /tr[1]/td[1][@bgcolor="#006699"]
15    /tr[1]/td[1][@bgcolor="#006699"]/font[1][@color="#ffffff"]
16    /tr[1]/td[1][@bgcolor="#006699"]/font[1][@color="#ffffff"]/t
17    ext()[1]
18    /tr[1]/td[1][@bgcolor="#006699"]/font[1][@color="#ffffff"]/b
19    [1]
20    /tr[2]
21    /tr[2]/td[1]
22    /tr[2]/td[1]/small[1]
```

1 /tr[2]/td[1]/small[1]/li[1]
2 /tr[2]/td[1]/small[1]/li[1]/a[1]
3 /tr[2]/td[1]/small[1]/li[1]/a[1]/text()[1]
4 /tr[2]/td[1]/small[1]/li[2]
5 /tr[2]/td[1]/small[1]/li[2]/a[1]
6 /tr[2]/td[1]/small[1]/li[2]/a[1]/text()[1]
7 ...
8 /tr[2]/td[1]/small[1]/li[6]
9 /tr[2]/td[1]/small[1]/li[6]/a[1]
10 /tr[2]/td[1]/small[1]/li[6]/a[1]/text()[1]
11 /tr[2]/td[1]/small[1]/li[6]/div[1][@align="right"]
12 /tr[2]/td[1]/small[1]/li[6]/div[1][@align="right"]/text()[1]
13 /tr[2]/td[1]/small[1]/li[6]/div[1][@align="right"]/a[1]
14 /tr[2]/td[1]/small[1]/li[6]/div[1][@align="right"]/text()[2]
15 /tr[2]/td[1]/small[1]/li[6]/div[1][@align="right"]/text()[2]

16 Moreover, in the case of combining the matching method of
17 dynamic annotation of this time with the conventional
18 matching method of dynamic annotation, XPathS owned by
19 another group and the XPathS generated from the matching
20 pattern can also be handled integrally by listing up all of
21 the described XPathS.

22 Measures for case where p, br and b tags and text nodes

1 occur randomly:
2 In some cases, the p, br and b tags and the text nodes occur
3 randomly in a body text or the like of a certain content.
4 In order to take measures for such a case, it is necessary
5 to generate a matching pattern capable of being matched with
6 the text even if the p, br and b tags are
7 increased/decreased. For this purpose, processing of
8 converting all of the p, br and b tags into "ANY" nodes in
9 the case where a series of the p, br and b tags occurs in
10 the target page and the past pages. Specifically, these
11 tags are utilized as normal expressions in an "ANY" matching
12 pattern.

13 Generation of format of existing tree matching
14 description language:

15 The original pattern description has been used this time for
16 explaining the present invention. This pattern description
17 can be converted into a pattern matching description
18 language equivalent thereto. However, this conversion
19 becomes troublesome and lowers the readability of the
20 description because the original tree structure cannot be
21 stored and strict description of the attributes is required,
22 and therefore, the above pattern matching description

1 language has not be used for the explanation. Accordingly,
2 a part of the method for converting the notation used this
3 time into the existing pattern matching language (relaxNG
4 format) will be introduced.

5 First, a pattern as below is considered.

```
6 <table width="168">
7   <tbody>
8     <pat:repeat>
9       <tr bgcolor="ffffff">
10        <td>
11          <small>
12            <a>
13              <pat:text pat:type="any">
14            </a>
15          </small>
16        </td>
17      </tr>
18    </pat:repeat>
19  </tbody>
20 </table>
```

21 A conversion example where the above pattern is
22 converted into the relaxNG format is shown below. Note that

1 the description of the attributes is partially omitted. The
2 relaxNG is designed to described the Schema of the entire
3 XML document, and therefore, is constructed to described a
4 pattern matched with all including root tags. Here, the
5 Schema is used for matching of the subtree. Therefore, as
6 the implementation, the processing will be performed by the
7 following two steps.

8 Step 1: Listing up all of the table tags in HTML

9 Step 2: Evaluating the tables one by one whether they
10 are matched with the matching pattern

11 The following sample is premised on the implementation as
12 described above. Note that the following is a description
13 example according to the relaxNG format.

14 <?xml version="1.0" ?>

15 <grammar xmlns="http://relaxng.org/ns/structure/0.9">

16 <start>

17 <element name="table">

18 <attribute name="width">

19 <value>168</value>

20 </attribute>

21 <zeroOrMore>

22 <choice>


```

1          <ref name="freeAttributesTABLE"/>
2      </choice>
3  </zeroOrMore>
4  <element name="tbody">
5      <zeroOrMore>
6          <choice>
7              <ref name="freeAttributresTBODY"/>
8          </choice>
9      </zeroOrMore>
10 <oneOrMore>
11     <element name="tr">
12         <attribute name="bgcolor">
13             <value>ffffff</value>
14         </attribute>
15         <zeroOrMore>
16             <choice>
17                 <ref name="freeAttributeTR"/>
18             </choice>
19         </zeroOrMore>
20         <element name="td">
21             <zeroOrMore>
22                 <choice>

```

```
1          <ref name="freeAttributesTD" />
2      </choice>
3  </zeroOrMore>
4  <element name="small">
5      <zeroOrMore>
6          <choice>
7              <ref name="freeAttributesSMALL" />
8          </choice>
9      </zeroOrMore>
10     <element name="a">
11         <zeroOrMore>
12             <choice>
13                 <ref name="freeAttributesA" />
14             </choice>
15         </zeroOrMore>
16         <text/>
17     </element>
18 </element>
19 </element>
20 </element>
21 </oneOrMore>
22 </element>
```

```
1      </element>
2      </start>

3      <define name="freeAttributesTD">
4      <attribute>
5          <anyName>
6              <except>
7                  <name>width</name>
8              </except>
9              <except>
10                 <name>height</name>
11             </except>
12 The rest is omitted.  In the TD tags, rows of attributes
13 unignorable in the matching is described here.
14         </anyName>
15     </attribute>
16 The rest is omitted.  freeAttributes definitions are arrayed
17 below for each tag.
18 </grammar>

19     Limitations on this method from viewpoint of matching
20 pattern generation capability:
21     It is known that two types, which are repeat and
```

1 embed, are present as latitude of the normal expression of
2 the tree. Between them, this method can detect only the
3 repeat. This is based on that the necessity of describing
4 the regularity by the embedding structure is extremely low
5 because the tree is used for matching the regions of the
6 HTML. Therefore, it is also possible to expand the matching
7 pattern to an algorithm calculating the embedding structure
8 based on a basic idea of using the statistical information.

9 [Other Example 1: Transcoding by Annotation]

10 It is possible to construct a "fail-safe" system
11 covering, by use of the free annotation of this time, leaked
12 pages, leaked information and the like as a result of
13 unmatching with the annotation in the conventional
14 annotation system. This greatly contributes to business
15 through quality assurance of the transcoding. Furthermore,
16 by performing the detailing of the matching conditions by
17 the present invention, labor for correcting the annotations
18 can be reduced, and the annotation authoring time can be
19 shortened. This is also a function that greatly contributes
20 to the business. Furthermore, the group portion that has
21 been able to be determined only by use of the character
22 string matching of the XPath in the conventional transcoding

1 can be covered by the free annotation. Fig. 32 shows an
2 example of a Web content where predetermined stationary
3 nodes move. In Fig. 32, portions such as "LYCOS Service"
4 and "Related Topics" sometimes move vertically, and are
5 difficult to handle by the conventional schema. This method
6 can also cope with such a group.

7 [Other Example 2: Generation of RSS by Cutout of Link List]

8 The RSS is called Rich Site Summary and is a standard for
9 enabling the summary of the site in various ways by defining
10 the summary of the site in the XML format and providing the
11 defined summary. Heretofore, the RSS has been dynamically
12 generated for each site by use of the CGI and the like.
13 However, by use of the present invention, it becomes
14 possible to dynamically generate the RSS from a Web page.
15 First, a free annotation that designates a link list serving
16 as a list of top news on a site is prepared by use of the
17 annotation editor. An "RSS attribute" is added to this
18 group. An RSS engine generates data in the RSS format
19 directly from the Web page by use of this free annotation.
20 It is difficult to realize a "group designating only a
21 specific portion" as described above by the conventional
22 annotation using the XPath matching. For example, in the

1 above-mentioned example of the pattern (XML format) with
2 reference to Fig. 20, the portions indicated in <pat:text
3 pat:type="any"> represent the respective titles of the top
4 articles of that day. Therefore, it becomes possible to
5 automatically generate RSS description as below by cutting
6 out wild card portions in the process of the pattern
7 matching.

```
8 <?xml version="1.0" encoding="utf-8" ?>
9 <rdf:RDF
10     xmlns="http://purl.org/rss/1.0/"
11     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
12     xm:lang="ja">
13     <channel rdf:about="http://news.lycos.co.jp/topics/rss.
14 rdf">
15         <title>News LYCOS Saishin Topics</title>
16         <link>http://news.lycos.co.jp/topics</link>
17         <items>
18             <rdf:Seq>
19                 <rdf:li
20 rdf:resource="http://news.lycos.co.jp/topics" />
21                 </rdf:Seq>
22         </items>
```

```
1  </channel>
2  <item
3  rdf:about="http://news.lycos.co.jp/topics/society/maff.
4  html">
5      <title>isahayawan kinpaku-no-naka koji saikai</title>
6      <link>http://news.lycos.co.jp/topics/society/maff.html
7  </link>
8  </item>
9  <item rdf:about="http://news.lycos.co.jp/topics/world/opera
10 -tion.html">
11      <title>arukaida sento-in amerika-ni-tokoh</title>
12      <link>http://news.lycos.co.jp/topics/world/operation.html
13 </link>
14 </item>
15 <item rdf:about="http://news.lycos.co.jp/topics/computer/ms.
16 html>
17      <title>maikurosofuto kadenbunya shinshutsu</title>
18      <link>http://news.lycos.co.jp/topics/computer/ms.html
19 </link>
20 </item>
21 ... (the rest is omitted)
22 </rdf:RDF>
```

1 [Other Example 3: Web Servicing of Web Page by Partial
2 Cutout]

3 The Web service is a technology of providing various
4 services and applications by an exchange of the XMLs. The
5 use of the present invention brings a possibility that the
6 services and the applications can be easily provided in a
7 way where the transaction of the existing Web page is
8 partially cut out. Fig. 33 shows an example of a Web
9 content to be used for the partial cutout. This page is a
10 page that submits a page including results of keyword search
11 for the past articles in a certain news site (ZDNET). A Web
12 service performing keyword search by use of this page as a
13 base can be constructed. Groups necessary to be designated
14 are two. one is a form portion 103 for the search (Fig.
15 33). This region is composed of unmoved portions, where the
16 matching pattern is easy to generate.

17 The next is one expressing the form portion 103 in the HTML.

18 <select name="idxname" size="1" tabindex="2">

19 <option value="" selected>ALL ZDNet

20 <option value="news">ZDNN

21 <option value="zdii">ZDII

22 ...


```
1  </select>
2      Next is a part (RelaxNG format) of Schema
3  automatically generated from the above-described HTML.  This
4  Schema is utilized as Schema (1) in Fig. 34.
5  <element name="idxname">
6  <choice>
7      <element name="option">
8      <element name="value">
9          <string></string>
10     </element>
11     </element>
12     <element name="option">
13     <element name="value">
14         <string>news</string>
15     </element>
16     </element>
17     <element name="option">
18     <element name="value">
19         <string>zdi</string>
20     </element>
21     </element>
22 </choice>
```

1 </element>

2 ...

3 Fig. 34 shows a processing course for automatically
4 generating the Web service from the Web content of Fig. 33.
5 For the cut out form, it is possible to automatically
6 generate XML Schema (Schema (1) of Fig. 34) for inputting
7 the cut out form and XSLT (XSLT (2)) for converting this XML
8 into an original HTML form.

```
9     <web_form_based_service action="./index.cgi" method="GET">
10         <text>kensaku (search) keyword</text>
11         <idxname>kensaku han-i shitei (search range
12 designation)</idxname>
13         <max>saidai kensaku kekka suu (maximum search result
14 number)</max>
15     </web_form_based_service>
```

16 Furthermore, it is necessary to change vocabularies
17 and to correct the automatically generated XSLT, XML Schema
18 and WSDL. However, it will be necessary to utilize the Web
19 service as a base for detailed development following
20 prototyping performed therefor. Although imperfect, the use
21 of the Web form in the way described above makes it possible
22 to prototype the Web service relatively easily. This has

1 been partially realizable also by a technology such as CHIP
2 heretofore.

3 The problem is the portion 104 of the search results
4 (Fig. 33). The portion 104 of the search results is a
5 portion where a varied content is dynamically generated, and
6 patterning for this portion is extremely difficult.
7 However, because the use of the present invention makes it
8 possible to determine the stationary nodes, the updated
9 nodes and the additional nodes, and further to detect the
10 repeated pattern, a pattern as below can be automatically
11 generated (cutout (5) by the pattern in Fig. 34). The next
12 pattern description corresponds to the cutout (5) by the
13 pattern in Fig. 34, and this pattern is not in RelaxNG but
14 in an original format.

15 <h2>kensaku kekka (search result)</h2>

16 <p>

17 sankou hit suu (reference hit number): [

18 <pat:text pat:type="any"/>

19 <pat:text pat:type="any" pat:format="[0-9] +"/>

20]

21 </p>

22 <p>

```
1      <strong>
2          <pat:text pat:type="any" pat:format="[0-9] +"/>
3          pieces of documents matched with search expression
4          were found.
5      </strong>
6  </p>
7  <dl>
8  <repeat>
9      <a>
10         <pat:text pat:type="any" />
11         <b>
12             <font color=blue>
13                 pat:text pat:type="any" />
14             </font>
15         </b>
16         <pat:text pat:type="any" />
17         <pat:text pat:type="inserted" />
18     </a>
19     font color=red size=-2>
20     (
21     <em>
22         <pat:text pat:type="any" />
```

```
1         </em>
2     )
3 </font>
4 <br>
5     <pat:text pat:type="any" />
6     <b>
7         <font color=blue>
8             <pat:text pat:type="any" />
9         </font>
10    </b>
11    <pat:text pat:type="any" />
12        <pat:text pat:type="inserted" />
13        <br>
14        <font color=green>
15            <pat:text pat:type="any" />
16        </font>
17        <br><br>
18 </repeat>
```

19 The cutout of the result portions can be performed form
20 this pattern, and the XML to be outputted can be generated
21 therefrom. Then, the repeated portions are itemized, and it
22 is possible to automatically generate the XML Schema ((4) in

1 Fig. 34) for outputting the updated portions except for the
2 repeated portions by special tags, the XSLT ((3) in Fig. 34)
3 for converting the HTML of the cut out portions into the XML
4 format, and the XSLT ((6) in Fig. 34) for decoding the XML
5 to the HTML.

6 [Other Example 4: Application to Information Aggregator]
7 Partial cut out of Web pages and integration of information
8 are broadly performed in a portal construction system such
9 as the IBM PortalServer and an information
10 extraction/submission system such as the IBM mySiteOutliner.
11 The present invention is applicable to these systems. For
12 example, in the IBM mySiteOutliner, XPath as below is held
13 in a definition file in order to extract a headline link
14 list from the Web page.

```
15     <ClippingDefinition>
16         <id>2</id>
17         <links>
18             <link title="Club IBM Top
19 Page>http://www.ibm.com/jp/pc/ clubibm/index.html</link>
20         </links>
21         <urldata>
22             <url source="Club IBM">http://www.ibm.com/jp/pc/
```

```
1 clubibm/index.html</url>
2     <xpathlists>
3         xpath name="body text">
4 /html[1]/body[1]/table[2]/tbody[1]/tr[1]/td[2]/table[2]/tbody[1]/tr[5]/td[2]/table[1]/tbody[1]/tr[1]/td[1]/table[2]/tbody[1]/tr[2]/td[1]
5
6         </xpath>
7     </xpathlists>
8 </urldata>
9 </ClippingDefinition>
```

11 The designation of the cut out portions depends on the
12 underlined XPath. Usually, the XPath format is weak against
13 the layout change, thus causing a problem of a large load in
14 maintenance. Specifically, a person must monitor the layout
15 change, and when there is a change, it is necessary to
16 manually author a correct XPath again. In the case of the
17 mySiteOutliner, the layout change is informed in advance
18 because a subject thereof is in-company page contents.
19 Therefore, the mySiteOutliner copes with the above-described
20 problem by delivering the XML file corrected simultaneously
21 with the layout change to the users. However, the problem
22 of the management cost is still present.

1 On the contrary, the application of the present
2 invention makes it possible to automatically generate the
3 matching pattern in a way below. This pattern uses, as
4 keys, contents of the subject table, and particularly, a
5 stationary character string such as "shincyoku-jyouhou
6 (What's New)" and an attribute of the table. Therefore,
7 this pattern is not shifted unless a change for these
8 character string and attribute occurs. The matching pattern
9 according to the present invention is excellent in that it
10 is completely robust for the insertion of the table
11 immediately under the body, the insertion of tr into the
12 upper table tag, which cause shifts under the current
13 circumstances, and the insertion of the div tag and the span
14 tag into the upper nodes, which does not cause a visual
15 influence.

16 <tr>

17 <td width="440" height="20" bgcolor="#CCCCCC"> nbsp;nbsp;

18 shincyoku-jyouhou (What's new) </td>

19 </tr>

20 <tr>

21 <td>

22 <table border="0" cellpadding="0" cellspacing="2">


```

1      <tbody>
2          <repeat>
3              <tr>
4                  <td>
5                      <pat:img pat:img_type="bullet"/>
6                  </td>
7                  <td>
8                      <a>
9                          <font color="#006699"><pat:text
10 pat:type="any" /> </font>
11                      </a>
12                  </td>
13              </tr>
14          </repeat>
15      </tbody>
16  </table>
17  </td>
18  </tr>

```

19 As cases where the robustness is lost in this pattern, for
20 example, cases as below are considered.

21 (a) Contents matched with the same pattern are inserted
22 into the same page.

1 (b) Attributes such as a background color and a font
2 color are changed.

3 Case (a) means that a region identical also visually occurs,
4 and the case is considered rare. For case (b), there is no
5 measure but generation of another pattern. However, in the
6 present invention, there is a possibility that a robust
7 pattern can be generated for both of the layouts (before and
8 after the layout change) by also using the pages before the
9 layout change for calculating the statistical quantity.
10 Therefore, the present invention can also cope with the
11 problem in the case (b).

12 In conclusion, the following items are disclosed
13 regarding the constitution of the present invention.

14 (1): A processing apparatus for a structured/hierarchical
15 content, which makes a determination whether or not a
16 structured/hierarchical content delivered through a network
17 includes a content portion matched with a predetermined
18 matching pattern, and performs predetermined processing for
19 the structured/hierarchical content if a result of the
20 determination is positive, the processing apparatus
21 includes: target subtree setting means for setting a target
22 subtree relating to a range including a target content

1 portion as an extracted portion of the matching pattern in
2 the structured/hierarchical content (hereinafter, referred
3 to as a "target content") from which the matching pattern is
4 to be extracted; occurrence mode detecting means for
5 detecting an occurrence mode of each node of the target
6 subtree by selecting a plurality of past
7 structured/hierarchical contents with respect to the target
8 content and collating the target subtree relating to the
9 target content with a tree relating to each of the past
10 structured/hierarchical contents; statistical information
11 generating means for generating statistical information
12 concerning an occurrence frequency of the occurrence mode of
13 each node in the target subtree based on the plurality of
14 past structured/hierarchical contents; classifying means for
15 performing classification of each node of the target subtree
16 based on the statistical information and a result of
17 detecting the occurrence mode; and matching pattern
18 generating means for generating the matching pattern for the
19 target content portion based on the classification.
20 (2): The processing apparatus for a structured/hierarchical
21 content according to (1) is characterized in that the
22 predetermined processing is to associate related information

1 with the content portion of the structured/hierarchical
2 content.

3 (3): The processing apparatus for a structured/hierarchical
4 content according to (2) is characterized in that the
5 related information includes an annotation.

6 (4): The processing apparatus for a structured/hierarchical
7 content according to (1) is characterized in that the
8 predetermined processing is processing for copying the
9 content portion of the structured/hierarchical content for a
10 purpose of utilizing the content portion of the
11 structured/hierarchical content for another
12 structured/hierarchical content.

13 (5): The processing apparatus for a structured/hierarchical
14 content according to any one of (1) to (4) is characterized
15 in that the structured/hierarchical content is a Web
16 content.

17 (6): The processing apparatus for a structured/hierarchical
18 content according to any one of (1) to (5) is characterized
19 in that the classifying means classifies nodes of the target
20 subtree into stationary nodes, updated nodes and additional
21 nodes.

22 (7): The processing apparatus for a structured/hierarchical

1 content according to claim (6) is characterized in that the
2 occurrence mode detecting means includes, as the occurrence
3 mode to be detected, (N1) an occurrence mode where detected
4 nodes occur in both of the target content portion and
5 structured/hierarchical contents collated therewith and
6 contents thereof are mutually identical, and (N2) an
7 occurrence mode where the detected nodes occur in both of
8 the target content portion and the structured/hierarchical
9 contents collated therewith and the contents thereof are
10 mutually different, and in that the classifying means
11 classifies, into the stationary nodes, nodes of which
12 occurrence frequency of the occurrence mode (N1) is
13 determined to be equal to/more than a first threshold value
14 by the statistical information, classifies, into the updated
15 nodes, nodes of which occurrence frequency of the occurrence
16 mode (N2) is determined to be equal to/more than a second
17 threshold value by the statistical information, and
18 classifies, into the additional nodes, nodes other than the
19 stationary nodes and the updated nodes.
20 (8): The processing apparatus for a structured/hierarchical
21 content according to any one of (6) and (7) is characterized
22 in that the matching pattern generating means includes:

1 repeated portion detecting means for detecting a repeated
2 portion in the target subtree based on the classification
3 into the stationary nodes, the updated nodes and the
4 additional nodes; and repeated information-added matching
5 pattern generating means for generating the matching pattern
6 including presence information of the repeated portion.
7 (9): The processing apparatus for a structured/hierarchical
8 content according to (8) is characterized in that the
9 classifying means includes: formed-for-spacer image
10 detecting means for detecting whether or not a node relating
11 to an image is a node relating to a formed-for-spacer image
12 for ensuring a blank region; bullet image detecting means
13 for detecting whether or not the node relating to the image
14 is a node relating to a plurality of bullet images used
15 repeatedly in a same size; first classifying means for
16 classifying the node relating to the formed-for-spacer image
17 into the additional nodes; and second classifying means for
18 allocating a plurality of the nodes relating to the bullet
19 image into a same classification among classifications of
20 the stationary nodes, updated nodes and additional nodes
21 even if display contents of the plurality of nodes are
22 mutually different.

1 (10): The processing apparatus for a structured/hierarchical
2 content according to any one of (1) to (9), further
3 includes: collating means for collating the target subtree
4 relating to the target content with the trees relating to a
5 plurality of structured/hierarchical contents adjacent to
6 the target content by selecting the adjacent
7 structured/hierarchical contents in place of the past
8 structured/hierarchical contents with respect to the target
9 content when the past structured/hierarchical contents are
10 not present.

11 (11): A processing apparatus for a structured/hierarchical
12 content, which makes a determination whether or not a
13 structured/hierarchical content delivered through a network
14 includes a content portion matched with a predetermined
15 matching pattern, and performs predetermined processing for
16 the structured/hierarchical content if a result of the
17 determination is positive, the processing apparatus
18 includes: target subtree setting means for setting a target
19 subtree relating to a range including a target content
20 portion as an extracted portion of the matching pattern in
21 the structured/hierarchical content (hereinafter, referred
22 to as a "target content") from which the matching pattern is

1 to be extracted; occurrence mode detecting means for
2 detecting an occurrence mode of each node of the target
3 subtree by selecting a plurality of structured/hierarchical
4 contents adjacent to the target content and collating the
5 target subtree relating to the target content with a tree
6 relating to each of the adjacent structured/hierarchical
7 contents; statistical information generating means for
8 generating statistical information concerning an occurrence
9 frequency of the occurrence mode of each node in the target
10 subtree based on the plurality of adjacent
11 structured/hierarchical contents; classifying means for
12 performing classification of each node of the target subtree
13 based on the statistical information and a result of
14 detecting the occurrence mode; and matching pattern
15 generating means for generating the matching pattern for the
16 target content portion based on the classification.
17 (12): A processing method for a structured/hierarchical
18 content, which makes a determination whether or not a
19 structured/hierarchical content delivered through a network
20 includes a content portion matched with a predetermined
21 matching pattern, and performs predetermined processing for
22 the structured/hierarchical content if a result of the

1 determination is positive, the processing method includes: a
2 target subtree setting step of setting a target subtree
3 relating to a range including a target content portion as an
4 extracted portion of the matching pattern in the
5 structured/hierarchical content (hereinafter, referred to as
6 a "target content") from which the matching pattern is to be
7 extracted; an occurrence mode detecting step of detecting an
8 occurrence mode of each node of the target subtree by
9 selecting a plurality of past structured/hierarchical
10 contents with respect to the target content and collating
11 the target subtree relating to the target content with a
12 tree relating to each of the past structured/hierarchical
13 contents; a statistical information generating step of
14 generating statistical information concerning an occurrence
15 frequency of the occurrence mode of each node in the target
16 subtree based on the plurality of past
17 structured/hierarchical contents; a classifying step of
18 performing classification of each node of the target subtree
19 based on the statistical information and a result of
20 detecting the occurrence mode; and a matching pattern
21 generating step of generating the matching pattern for the
22 target content portion based on the classification.

1 (13): The processing method for a structured/hierarchical
2 content according to (12) is characterized in that the
3 predetermined processing is to associate related information
4 with the content portion of the structured/hierarchical
5 content.

6 (14): The processing method for a structured/hierarchical
7 content according to (13) is characterized in that the
8 related information includes an annotation.

9 (15): The processing method for a structured/hierarchical
10 content according to (12) is characterized in that the
11 predetermined processing is processing for copying the
12 content portion of the structured/hierarchical content for a
13 purpose of utilizing the content portion of the
14 structured/hierarchical content for another
15 structured/hierarchical content.

16 (16): The processing method for a structured/hierarchical
17 content according to any one of (12) to (15) is
18 characterized in that the structured/hierarchical content is
19 a Web content.

20 (17): The processing method for a structured/hierarchical
21 content according to any one of (12) to (16) is
22 characterized in that in the classifying step, nodes of the

1 target subtree are classified into stationary nodes, updated
2 nodes and additional nodes.

3 (18): The processing method for a structured/hierarchical
4 content according to (17) is characterized in that the
5 occurrence mode detecting step includes, as the occurrence
6 mode to be detected, (N1) an occurrence mode where detected
7 nodes occur in both of the target content portion and
8 structured/hierarchical contents collated therewith and
9 contents thereof are mutually identical, and (N2) an
10 occurrence mode where the detected nodes occur in both of
11 the target content portion and the structured/hierarchical
12 contents collated therewith and the contents thereof are
13 mutually different, and in that in the classifying step, are
14 classified into the stationary nodes, nodes of which
15 occurrence frequency of the occurrence mode (N1) is
16 determined to be equal to/more than a first threshold value
17 by the statistical information, are classified into the
18 updated nodes, nodes of which occurrence frequency of the
19 occurrence mode (N2) is determined to be equal to/more than
20 a second threshold value by the statistical information, and
21 are classified into the additional nodes, nodes other than
22 the stationary nodes and the updated nodes.

1 (19): The processing method for a structured/hierarchical
2 content according to any one of (17) and (18) is
3 characterized in that the matching pattern generating step
4 includes: a repeated portion detecting step of detecting a
5 repeated portion in the target subtree based on the
6 classification into the stationary nodes, the updated nodes
7 and the additional nodes; and a repeated information-added
8 matching pattern generating step of generating the matching
9 pattern including presence information of the repeated
10 portion.

11 (20): The processing method for a structured/hierarchical
12 content according to (19) is characterized in that the
13 classifying step includes: a formed-for-spacer image
14 detecting step of detecting whether or not a node relating
15 to an image is a node relating to a formed-for-spacer image
16 for ensuring a blank region; a bullet image detecting step
17 of detecting whether or not the node relating to the image
18 is a node relating to a plurality of bullet images used
19 repeatedly in a same size; a first classifying step of
20 classifying the node relating to the formed-for-spacer image
21 into the additional nodes; and a second classifying step of
22 allocating a plurality of the nodes relating to the bullet

1 image into a same classification among classifications of
2 the stationary nodes, updated nodes and additional nodes
3 even if display contents of the plurality of nodes are
4 mutually different.

5 (21): The processing method for a structured/hierarchical
6 content according to any one of (12) to (20), further
7 includes: a collating step of collating the target subtree
8 relating to the target content with the trees relating to a
9 plurality of structured/hierarchical contents adjacent to
10 the target content by selecting the adjacent
11 structured/hierarchical contents in place of the past
12 structured/hierarchical contents with respect to the target
13 content when the past structured/hierarchical contents are
14 not present.

15 (22): A processing method for a structured/hierarchical
16 content, which makes a determination whether or not a
17 structured/hierarchical content delivered through a network
18 includes a content portion matched with a predetermined
19 matching pattern, and performs predetermined processing for
20 the structured/hierarchical content if a result of the
21 determination is positive, the processing method includes: a
22 target subtree setting step of setting a target subtree

1 relating to a range including a target content portion as an
2 extracted portion of the matching pattern in the
3 structured/hierarchical content (hereinafter, referred to as
4 a "target content") from which the matching pattern is to be
5 extracted; an occurrence mode detecting step of detecting an
6 occurrence mode of each node of the target subtree by
7 selecting a plurality of structured/hierarchical contents
8 adjacent to the target content and collating the target
9 subtree relating to the target content with a tree relating
10 to each of the adjacent structured/hierarchical contents; a
11 statistical information generating step of generating
12 statistical information concerning an occurrence frequency
13 of the occurrence mode of each node in the target subtree
14 based on the plurality of adjacent structured/hierarchical
15 contents; a classifying step of performing classification of
16 each node of the target subtree based on the statistical
17 information and a result of detecting the occurrence mode;
18 and a matching pattern generating step of generating the
19 matching pattern for the target content portion based on the
20 classification.

21 (23): A program for allowing a computer to execute the steps
22 of the processing method for a structured/hierarchical

1 content according to any one of (12) to (22).

2 [Advantages of the invention]

3 In the present invention, not an XPath but a matching
4 pattern is used in order to identify whether or not a
5 structured/hierarchical content is to be subjected to
6 processing such as partial cutout and reuse of a common
7 annotation. Consequently, the present invention can
8 flexibly cope with a case where the identified content
9 portion moves in the structured/hierarchical content to be
10 identified.

11 In the present invention, past and/or adjacent
12 structured/hierarchical contents with respect to a target
13 content are checked, and each node in a target subtree is
14 classified based on an occurrence mode of each node and
15 statistical information concerning an occurrence frequency
16 of the occurrence mode, and thus the matching pattern is
17 generated. Consequently, a matching pattern, which is
18 significant for identifying whether or not the
19 structured/hierarchical content is to be subjected to the
20 processing, can be generated.

21 Although the preferred embodiments of the present
22 invention have been described in detail, it should be

1 understood that various changes, substitutions and
2 alternations can be made therein without departing from
3 spirit and scope of the inventions as defined by the
4 appended claims. Variations described for the present
5 invention can be realized in any combination desirable for
6 each particular application. Thus particular limitations,
7 and/or embodiment enhancements described herein, which may
8 have particular advantages to the particular application
9 need not be used for all applications. Also, not all
10 limitations need be implemented in methods, systems and/or
11 apparatus including one or more concepts of the present
12 invention.

13 The present invention can be realized in hardware,
14 software, or a combination of hardware and software. A
15 visualization tool according to the present invention can be
16 realized in a centralized fashion in one computer system, or
17 in a distributed fashion where different elements are spread
18 across several interconnected computer systems. Any kind of
19 computer system - or other apparatus adapted for carrying
20 out the methods and/or functions described herein - is
21 suitable. A typical combination of hardware and software
22 could be a general purpose computer system with a computer

1 program that, when being loaded and executed, controls the
2 computer system such that it carries out the methods
3 described herein. The present invention can also be
4 embedded in a computer program product, which comprises all
5 the features enabling the implementation of the methods
6 described herein, and which - when loaded in a computer
7 system - is able to carry out these methods.

8 Computer program means or computer program in the present
9 context include any expression, in any language, code or notation,
10 of a set of instructions intended to cause a system having an
11 information processing capability to perform a particular function
12 either directly or after conversion to another language, code or
13 notation, and/or reproduction in a different material form.

14 Thus the invention includes an article of manufacture
15 which comprises a computer usable medium having computer
16 readable program code means embodied therein for causing a
17 function described above. The computer readable program
18 code means in the article of manufacture comprises computer
19 readable program code means for causing a computer to effect
20 the steps of a method of this invention. Similarly, the
21 present invention may be implemented as a computer program
22 product comprising a computer usable medium having computer
23 readable program code means embodied therein for causing a a

1 function described above. The computer readable program
2 code means in the computer program product comprising
3 computer readable program code means for causing a computer
4 to effect one or more functions of this invention.
5 Furthermore, the present invention may be implemented as a
6 program storage device readable by machine, tangibly
7 embodying a program of instructions executable by the
8 machine to perform method steps for causing one or more
9 functions of this invention.

10 It is noted that the foregoing has outlined some of the
11 more pertinent objects and embodiments of the present invention.
12 This invention may be used for many applications. Thus,
13 although the description is made for particular arrangements
14 and methods, the intent and concept of the invention is
15 suitable and applicable to other arrangements and
16 applications. It will be clear to those skilled in the art
17 that modifications to the disclosed embodiments can be
18 effected without departing from the spirit and scope of the
19 invention. The described embodiments ought to be construed to be
20 merely illustrative of some of the more prominent features and
21 applications of the invention. Other beneficial results can be
22 realized by applying the disclosed invention in a different manner
23 or modifying the invention in ways known to those familiar with the